ERDC/CRREL TR-14-20

US Army Corps
of Engineers®
Engineer Research and
Development Center

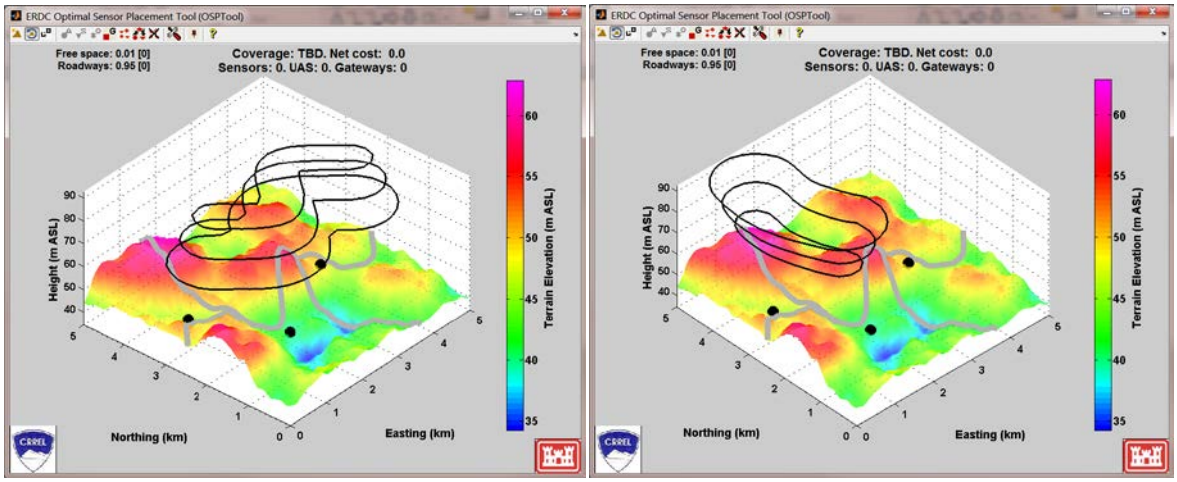
ERDC
INNOVATIVE SOLUTIONS
for a safer, better world

AT42 GRE Geospatially Enabled Multimodal Situation Awareness—Ground and Air Surveillance Optimization (GEMS-GASO)

Coordinated Optimization of Aircraft Routes and Locations of Ground Sensors

Sergey N. Vecherin, D. Keith Wilson, and Chris L. Pettit

September 2014



Cold Regions Research and Engineering Laboratory

The U.S. Army Engineer Research and Development Center (ERDC) solves the nation's toughest engineering and environmental challenges. ERDC develops innovative solutions in civil and military engineering, geospatial sciences, water resources, and environmental sciences for the Army, the Department of Defense, civilian agencies, and our nation's public good. Find out more at www.erdc.usace.army.mil.

To search for other technical reports published by ERDC, visit the ERDC online library at http://acwc.sdp.sirsi.net/client/default.

ERDC/CRREL TR-14-20
September 2014

Coordinated Optimization of Aircraft Routes and Locations of Ground Sensors

Sergey N. Vecherin and D. Keith Wilson

Cold Regions Research and Engineering Laboratory (CRREL)
U.S. Army Engineer Research and Development Center
72 Lyme Road
Hanover, NH 03755-1290

Chris L. Pettit

U.S. Naval Academy
590 Holloway Rd., MS 11-B
Annapolis, MD 21402

Final Report

Prepared for Headquarters, U.S. Army Corps of Engineers
Washington, DC 20314-1000

Under AT42 GRE Geospatially Enabled Multimodal Situation Awareness—Ground and Air Surveillance Optimization (GEMS-GASO)

Abstract

In this report, an optimal sensor placement tool, developed for determining near-optimal configurations of stationary ground sensors, is generalized to support aircraft routing. This generalization requires characterizing candidate aircraft routes in terms of cost and coverage. Cost can reflect a variety of disincentives, not necessarily monetary—for example, a probability of aircraft to be heard on the ground. Several metrics for moving sensor platforms were considered to adequately characterize cost and coverage. The generalized algorithm can be applied to such practical problems as determining the optimal combination of routes for multiple aircraft operations, optimizing routes to supplement ground-sensor coverage, optimizing ground sensors to cover blind spots of aircraft coverage, and simultaneously optimizing static and moving sensor platforms. An example problem that this report considers in detail is unmanned aircraft system (UAS) routing for verification of roadway security while minimizing UAS audibility at specified locations on the ground.

Contents

Illustrations

Figures

Preface

This study was conducted for the U.S. Army Corps of Engineers, Engineer Research and Development Center, under the AT42 Geospatial Research and Engineering project as a part of Geospatially Enabled Multimodal Situation Awareness—Ground and Air Surveillance Optimization (GEMS-GASO).

The work was performed by Dr. Sergey Vecherin and Dr. Keith Wilson (Signature Physics Branch, Dr. Lindamae Peck, Chief), U.S. Army Engineer Research and Development Center, Cold Regions Research and Engineering Laboratory (ERDC-CRREL), and Dr. Chris Pettit (U.S. Naval Academy). At the time of publication, Dr. Justin Berman was Chief of the Research and Engineering Division. The Deputy Director of ERDC-CRREL was Dr. Lance Hansen, and the Director was Dr. Robert Davis.

COL Jeffrey R. Eckstein was the Commander of ERDC, and Dr. Jeffery P. Holland was the Director.

1 Introduction

The problem of optimal coverage for ground sensors is to determine the numbers, types, and locations of sensors needed to satisfy coverage preferences, specified by a mission planner, with minimal total cost. Such a formulation is a generalization of the original art gallery problem, which is to determine the number and locations of the guards so that each painting in the gallery is seen by at least one guard (O'Rourke 1987). The generalization manifests in two respects. First, simultaneous optimization of multiple types of sensors is supported; and second, sensors may not necessarily be line-of-sight. Vecherin et al. (2010, 2011) present an exact problem formulation with these modifications and such constraints as limited sensor supply, multiple sensor coverage, and wireless sensor communication. Mathematically, the problem falls into a category of the nondeterministic-polynomial (NP-hard) problems (Sierksma 2002) and, therefore, requires significant computational resources; for typical mission planning tasks, exact solution algorithms become impractical. To overcome this issue, a greedy algorithm had been developed yielding a suboptimal solution. The main idea of the algorithm is to place sensors one-by-one at locations that provide the maximum of overall coverage. This algorithm is a generalization of the greedy algorithm for minimal set cover (Johnson 1974), known in combinatorial mathematics, to support the probabilistic sensor performance framework. According to Feige (1998), such greedy algorithms are optimal among other possible approximate algorithms, considering the proximity of the provided solution to the exact solution and the elapsed computational time. Multiple example scenarios considered by Vecherin et al. (2010, 2011) and comparison with other heuristic and exact algorithms for optimal sensor placement confirm high efficiency and versatility of that approach.

The goal of this report is to device an approach for generalizing the approximate stationary sensor placement algorithm to support optimization of moving sensor platforms, such as aircraft in general and unmanned aircraft systems (UAS) in particular. If such a generalization is possible, the same computationally efficient approximate algorithm would enable solutions to such problems as determining an optimal route (or combination of routes) for airborne sensing platforms, determining a route to complement coverage provided by stationary ground sensors, determining an optimal

ground-sensor configuration to cover blind spots of available UAS coverage, and simultaneously optimizing UAS routes and types and locations of ground sensors.

In general, optimal routing is desirable in many areas; and there is a solid body of literature describing optimization for different purposes. The approaches for the solution can be grouped into three large classes: (1) approaches operating in continuous space and time; (2) approaches using structured space and time, such as optimization on graphs; and (3) heuristic approaches, for example, simulating annealing, ant colony, tabu search, and genetic algorithms.

The approaches of the first class are exemplified by a classic work of Dubins (1957) who considered solutions for the shortest path for a vehicle with a constant speed, finite maximal curvature, and specified initial and terminal positions and tangents of the route. Analysis of the solutions revealed that all such routes in the two-dimensional Euclidean space (for example, a plane at a constant altitude) consist of no more than three smoothly connected segments, each of which is either a straight line segment or an arc of the maximal curvature. The first class' approaches are also common in the optimal control and planning areas (Boscain and Piccoli 2004; LaValle 2006) and in computational geometry, where this problem is often referred to as the watchman routing problem (e.g., Chin and Ntafos 1991; Wang et al. 2007), although solution methods are substantially distinct in these two areas. Khardi and Abdallah (2012) applied the variational calculus approach to determine an optimal descending path and throttle control to minimize sound levels at certain locations around an airport. There are two major challenges in practical application of such approaches. First, some practically important constraints may be extremely difficult to formulate in such a framework; and second, an exact solution may require extensive computational resources and become impractical.

The approaches of the second class use the fact that any route, including the optimal one, must connect only a finite set of specified locations in the space that can be viewed as nodes on the graph, which brings the problem in the field of combinatorial optimization. Perhaps, the most renowned algorithm for optimal routing minimizing a sum of non-negative costs associated with the movement from one node to another is the Dijkstra algorithm (Dijkstra 1959). There are many modifications of this algorithm suitable for a variety of constraints (e.g., A* algorithm [Hart et al. 1968] is

widely used, which improves the efficiency [the speed of convergence] of Dijkstra's algorithm if the lower bound for the total cost is provided).

Route optimization approaches allowing minimization of the total cost subject to linear inequality constraints can be formulated in terms of the binary linear problem (Toth and Vigo 2001). A generic problem statement for one type of vehicles involves two-index logical variables, x_{ij}, that indicate arcs connecting one node with another. The optimal solution is given in terms of these variables, with possible values 1 or 0, indicating which arcs should be selected in the optimal route. Other constraints in the problem may be imposed to guarantee that the route is continuous (there are no disconnected arcs) and closed (terminates at the same node where it began). In such formulation, an elementary logic element is an arc, not a node, which is a principal distinction from the single-index optimal sensor formulation. Additional constraints can be incorporated in such a framework, such as visiting certain nodes several times and desired time windows for the visits.

The approaches in the third class represent universal optimizers, frequently inspired by biological or other natural phenomena, and their efficiency for a specific optimization problem depends on fortunate insights into physical or mathematical aspects of that problem. For example, simulated annealing (Osman 1993), genetic (Ombuki et al. 2006), ant colony (Bell and McMullen 2004), and tabu search (Gendreau et al. 1994) algorithms are frequently used. The last can also be used in the former metaheuristic algorithms to avoid convergence to local optima. Main advantages of such approaches are their robustness and ability to incorporate complex constraints.

Among all considered approaches, only one is fully compatible with the single-index binary linear programming formulation used for stationary ground-sensor optimization and, thus, is the most suitable for our research goal. In Toth and Vigo (2001, 21–22), it is referred to as the set-covering formulation. In such an approach, an elementary logic element is an entire vehicle route from a class of admissible routes (that is, routes satisfying other constraints). And a single-index logical variable indicates which of the admissible routes is the most optimal. Thus, mathematically, it is fully compatible with the stationary sensor selection and placement optimization, where a logical single-index variable indicates types and locations of sensors.

To this end, candidate aircraft routes should be generated and characterized in terms of provided coverage and associated cost, similarly to stationary ground sensors. As we will show below, there are a few ways of doing so; and each way may be suitable for certain missions.

2 Problem Statement

The set-covering approach to routing optimization is based on the same principles as the optimization of stationary ground sensors considered in Vecherin et al. (2010, 2011). First, coverage preferences are specified for the mission by indicating a desired level of probability of detection (at a certain fixed level of probability of false alarm) for each spatial location of the scene. Second, a set of candidate aircraft routes is created either with existing flight planning software, manually by a mission planner, or with the use of the algorithm considered in Section 4 of this report. Third, each candidate route is evaluated in terms of provided coverage and associated costs. Examples of what cost might be and nuances of cost characterizing entire aircraft routes are considered below. Finally, the most optimal combination of routes is chosen from among the candidates after evaluating cost-coverage relationships for each combination. Mathematically, such a determination of the most optimal combination out of many candidates can be strictly formulated in terms of the single-index binary linear programming problem outlined below.

Suppose, there are Q candidate aircraft routes, R_q, $q = 1,...,Q$. Assuming that distinct routes provide independent observations, the problem of determining the optimal combination of the routes can be stated as follows:

$$\begin{aligned} &\mathbf{u}_0 = \arg\min \mathbf{c}^T\mathbf{u}, \; u_q = \{0,1\}, \\ &\mathbf{A}\mathbf{u} \le \mathbf{b}, \end{aligned} \tag{1}$$

where

$\mathbf{u}$ = the indicative column vector whose elements can be either 0 or 1. Value 1 in the q* position of vector $\mathbf{u}$ (and $\mathbf{u}_0$) indicates that the route R_{q^*} should be taken,

$\mathbf{c}$ = the cost column vector,

T = the transpose operation,

$\mathbf{u}_0$ = the optimal vector minimizing the total cost,

$\mathbf{A}$ = the coverage matrix with elements $A_{iq} = \ln\left(1 - P_d(\mathbf{r}_i, \mathrm{R}_q)\right)$ characterizing the coverage at a spatial point, $\mathbf{r}_i$, provided by a

sensor moving along the candidate route R_q; P_d is the probability of detection,

b = the column-vector of given coverage preferences for each spatial point, $b_i = \ln\left(1 - P_d^{pref}(\mathbf{r}_i)\right)$, i = 1,...,N, where N indicates the total number of spatial points where the coverage is required. P_d^{pref} is the threshold probability of detection reflecting coverage preferences.

A specific distribution of ones in the indicative vector **u** yields a specific route combination. Analogously to the stationary ground sensors, the set of candidate routes may include subsets of the routes for distinct aircraft types: $Q = Q_1 + Q_2 + \ldots$. In this case, the solution will yield the number, the aircraft type, and corresponding optimal aircraft routes.

The cost may reflect any additive quantity and is determined by a mission goal. For example, the cost may be the actual cost to operate an aircraft; then, the goal of the optimization would be to minimize the total financial expenses for a specified mission. Another example is an abstract cost, **c** = [1, 1, 1, ...]; and the optimization would yield the minimal number of aircraft to accomplish the mission.

For example, let us consider a mission when aircraft should provide surveillance of a certain area on the ground with minimal audible disturbance at specified locations where hostile observers may be located. Practical examples include monitoring a stationary object (a building, hostile or friendly base camps, etc.) or roadways while avoiding audible detection by the observers. The locations of the observers may be within the object being observed (like in the case of monitoring a hostile base camp, for example) or somewhere else (for instance, when monitoring a friendly base camp or roadway). In some cases, the audibility of the aircraft at some locations should be, oppositely, maximized so that the enemies would detect the aircraft on purpose, which may prevent them from malicious actions.

In these scenarios, the cost should reflect the probability of aircraft being detected. Depending on a mission, a characterization of the entire aircraft route, R_q, in terms of provided coverage and probable detection may include several options considered in the next section.

3 Characterization of an Aircraft Route in Probabilistic Terms

The inherent difficulty in probabilistic characterization of an aircraft route is that the probabilities depend on the aircraft position, $\mathbf{s} = (s_x, s_y, s_z)$, changing with time. Indeed, at any given time, *t*, the aircraft position can be expressed as follows:

$$\mathbf{s}(t) = \mathbf{s}_0 + \int_0^t \mathbf{V}(t')dt', \tag{2}$$

where $\mathbf{s}_0$ is the starting aircraft position and $\mathbf{V} = (v_x, v_y, v_z)$ is the aircraft velocity, which is necessarily a function of time for any closed aircraft route. For example, consider an idealized closed route, a perfect circle with a radius *a* at a fixed altitude, $v_z = 0$. Even at the constant aircraft speed, $V = |\mathbf{V}| = const$, its horizontal components are time-dependent: $v_x = -V \sin \omega t$ and $v_y = V \cos \omega t$, where $\omega = V / a$ is the angular velocity.

As a result, coverage provided by aircraft and its detectability are also changing with time. In practice, time can be discretized by a finite set of time moments, $t = t_1, \ldots, t_M$, so that the entire route, R_q, can be presented by a set of *M* route points, $\mathrm{R}_q = \{\mathbf{s}_{1q}, \ldots, \mathbf{s}_{Mq}\}$, where $\mathbf{s}_{pq} = \mathbf{s}_q(t_p)$, $p = 1, \ldots, M$. These route points can be viewed as locations of individual sensors. Depending on a mission, several choices are feasible to characterize an aircraft route in probabilistic terms. Two typical missions are considered here:

Mission 1. Confirm or deny that a certain target is present in a certain location. To fulfill this mission, it is sufficient to detect the target from at least one route point. The corresponding probability of detection for this mission, $P_d^{(1)}(\mathbf{r})$, can be expressed in terms of the probability that a target will not be detected from all route points (missed detection) along route *q*, as follows:

$$P_d^{(1)}(\mathbf{r}) = 1 - P_{md}(\mathbf{r}; \mathbf{s}_{1q}, \ldots, \mathbf{s}_{Mq}), \tag{3}$$

where the superscript (1) denotes "at least one" and the semicolon separates the spatial point at which the probability of missed detection is calculated from the set of aircraft route points.

Mission 2. Constantly monitor a target or a certain area, such as a roadway. To fulfill this mission, the object of interest should be detected from all route points. The corresponding metric for this mission is the joint probability of detection from all route points along route q, $P_d(\mathbf{r};\mathbf{s}_{1q},\ldots,\mathbf{s}_{Mq})$.

A variety of other missions for UASs is possible, each of which requires its own unique objective function for optimization and constraints. For example, Measure et al. (2009) consider the problem of UAS optimal routing from a specified starting location to a specified destination with the primary focus on minimizing adverse weather effects on the UAS, such as rain, fog, ice, etc. These effects are characterized by a separate software called Tri-Service Integrated Weather Effects Decision Aid (T-IWEDA) that was previously developed by the Army Research Laboratory. Originally, these effects are characterized by Boolean values as "red," "yellow," and "green" hazardous zones. Analyzing the zone's thresholds derives a cost function, which is subject to minimization. Another optimization objective is considered in Lee (2012) for a multiple UASs employed in counterinsurgency operations. In this scenario, multiple target locations are specified in advance with the requirements to visit some of the locations once and some other locations twice. For each such visit, a score had been prescribed by a mission planner group, along with other mission-specific constraints, such as maintenance schedule, preferred time windows for observations, and time during which a target was observed. The objective function subject to minimization incorporates these scores. The set of the optimization objective functions and constraints in the problem determines an approach to solve it. In both these examples, probability of detection is not the objective of the optimization; and, furthermore, the objectives and constraints are specified by either a mission planner or calculated using separate software.

In our study, both missions stated above require joint probabilities (either missed detection or detection) for all route points. They can be easily calculated from individual route point probabilities only assuming independent observations. As shown below, for statistically dependent observations, the calculations are not trivial.

3.1 Coverage matrix for aircraft

Coverage matrix **A** enters the inequality coverage constraint in equation (1). For Mission 1, its elements are

$$A_{iq} = \ln\left(1 - P_d^{(1)}(\mathbf{r}_i)\right) = \ln P_{md}(\mathbf{r};\mathbf{s}_{1q},...,\mathbf{s}_{Mq}),$$

whereas, for Mission 2,

$$A_{iq} = \ln\left(1 - P_d(\mathbf{r}_i;\mathbf{s}_{1q},...,\mathbf{s}_{Mq})\right).$$

Note that, contrary to univariate probabilities, for multivariate probabilities (both dependent and independent),

$$P_{md}(\mathbf{r};\mathbf{s}_{1q},...,\mathbf{s}_{Mq}) + P_d(\mathbf{r};\mathbf{s}_{1q},...,\mathbf{s}_{Mq}) \neq 1$$

for any q. To calculate the joint probabilities, sensor information can be fused from all M observations (route points) either in the probability space or in the signal space.

3.1.1 Information fusion in the probability space

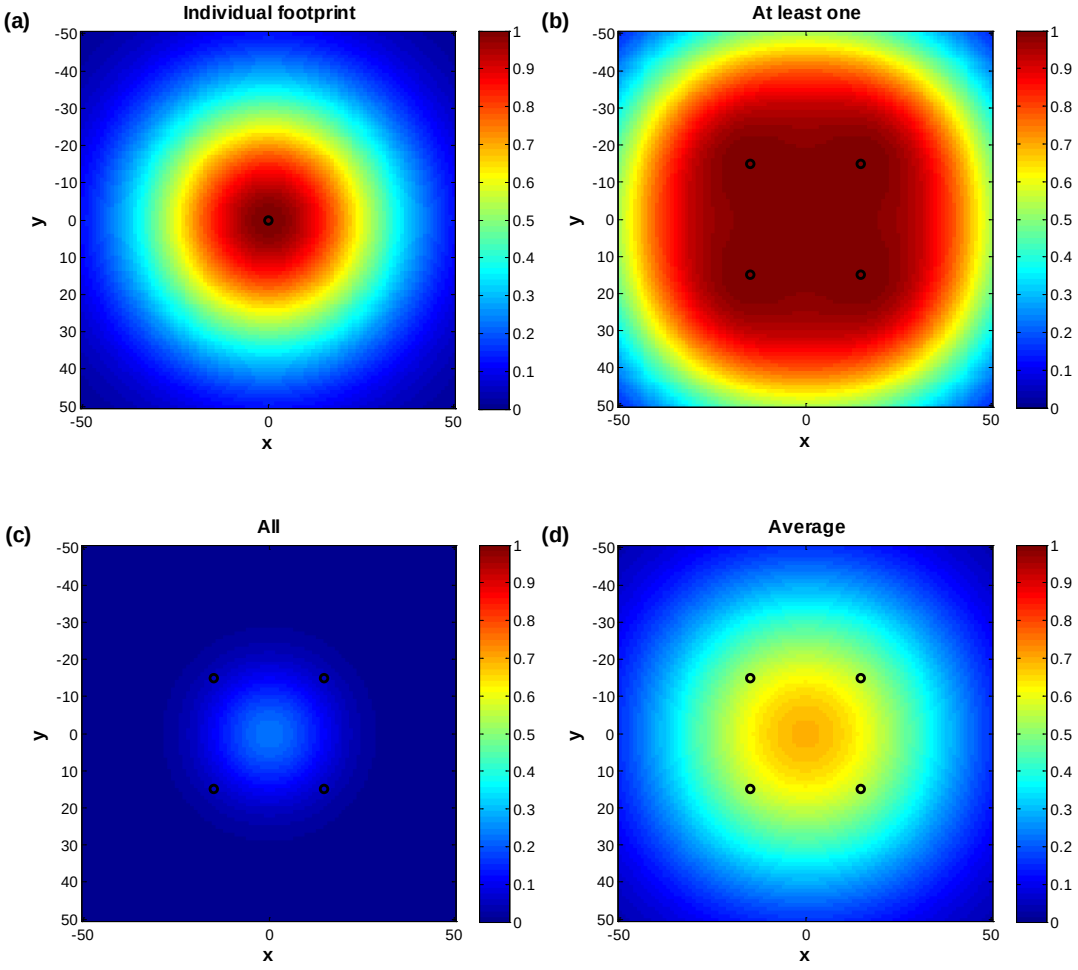
In this case, the probability of detection for each spatial location is given for each individual route point along the route, $P_d(\mathbf{r};\mathbf{s}_{pq})$. These probabilities are calculated at a certain threshold level of the false alarm probability, $P_{fa}(\mathbf{s}_{pq})$, also evaluated individually for each route point. If sensor readings are assumed to be independent, the probabilities for all route points equal the product of corresponding probabilities at every route point:

$$\begin{aligned} &P_{md}(\mathbf{r};\mathbf{s}_{1q},\ldots,\mathbf{s}_{Mq}) = \prod_{p=1}^{M} P_{md}(\mathbf{r};\mathbf{s}_{pq}), \quad P_d(\mathbf{r};\mathbf{s}_{1q},\ldots,\mathbf{s}_{Mq}) = \prod_{p=1}^{M} P_d(\mathbf{r};\mathbf{s}_{pq}), \\ &P_{fa}(\mathbf{s}_{1q},\ldots,\mathbf{s}_{Mq}) = \prod_{p=1}^{M} P_{fa}(\mathbf{s}_{pq}). \end{aligned} \tag{4}$$

Note that the false alarm probability for the entire set of observations is not equal to the original false alarm threshold, set for an individual route point. It decreases, along with the joint probabilities of detection and joint probabilities of missed detection. Therefore, the assumption of independ-

ent observations represents the best-case scenario for Mission 1, and the worst-case scenario for Mission 2. Figure 1 illustrates these detection probabilities assuming no obstacles and flat terrain. Figure 1(a) shows a sensing footprint from a single route point indicated by a black circle in the center. Of course, in complex terrain conditions, the footprints will be of irregular shapes; but the tendency, illustrated here, will remain the same.

Figure 1. Detection probabilities under different assumptions: (a) probability of detection on the ground from a single route point, indicated by a circle in the center as viewed from above; (b) probability of detection from at least one out of four route points; (c) probability of detection from all four route points; (d) average probability of detection over four route points.



The considered example assumes that all route points (black circles in all plots of Figure 1) have identical sensing footprints shown in Figure 1(a). Figure 1(b) and (c) depict the probabilities of detection for Mission 1, $P_d^{(1)}(\mathbf{r})$, and Mission 2, $P_d(\mathbf{r};\mathbf{s}_{1q},\ldots,\mathbf{s}_{Mq})$, respectively. Note a remarkable difference in covered area for these cases. These probabilities represent

extreme cases, not quite characterizing typical coverage along the route. To get an insight about average coverage, Figure 1(d) shows the averaged detection probability (over four route points). However, the caveat is that it does not really equal any instantaneous probability of detection. For example, if an object is hidden behind a wall and an aircraft circles around it, the detection of that object would have zero probability for one half of the route and unit probability for the other. But the route-averaged value would be 0.5.

The assumption of independent readings is very convenient but may not always be valid for arbitrary locations $\mathbf{s}_{pq}$. Apparently, for short periods of time, two consecutive route points practically coincide with each other; and the corresponding sensor readings cannot be treated as independent. Some route points can be close to each other due to route geometry even if they are significantly separated in time. For these cases, the chain rule for the dependent probabilities reads as follows:

$$P_{md}(\mathbf{r};\mathbf{s}_1,\ldots,\mathbf{s}_M) = P_{md}(\mathbf{r};\mathbf{s}_1)P_{md}(\mathbf{r};\mathbf{s}_2\,|\,\mathbf{s}_1)P_{md}(\mathbf{r};\mathbf{s}_3\,|\,\mathbf{s}_2,\mathbf{s}_1)\ldots P_{md}(\mathbf{r};\mathbf{s}_M\,|\,\mathbf{s}_{M-1},\ldots,\mathbf{s}_1), \quad (5)$$

but the conditional probabilities in the right-hand side of equation (5) are unknown and cannot be calculated based on the individual probabilities $P_{md}(\mathbf{r};\mathbf{r}_{pq})$. A more accurate approach to deal with dependent observations is to fuse information in the signal space.

3.1.2 Information fusion in the signal space

Let us review how the probability of detection at $\mathbf{r}$ is calculated for a single stationary sensor located at $\mathbf{s}$ (Burdic 1984):

$$P_d(\mathbf{r};\mathbf{s}) = \int_{\gamma}^{\infty} p_{U(\mathbf{r};\mathbf{s})}(w)dw\,, \quad (6)$$

where γ is the detection threshold determined from the following equation for some given P_{fa}:

$$P_{fa} = \int_{\gamma}^{\infty} p_{N(\mathbf{s})}(w)dw, \quad (7)$$

and $p_{N(\mathbf{s})}$ and $p_{U(\mathbf{r};\mathbf{s})}$ are probability density functions of a background noise and a received signal powers. The probability of missed detection for this univariate case is

$$P_{md}(\mathbf{r};\mathbf{s}) = 1 - P_d(\mathbf{r};\mathbf{s}) = \int_{-\infty}^{\gamma} p_{U(\mathbf{r};\mathbf{s})}(w)dw \,. \tag{8}$$

For a set of dependent observations, equations (6–8) can be generalized as follows. First, depending on mission objectives, the specified detection threshold of a false alarm may be applied not only to an individual observation but also to the entire set of observations. Let us denote that threshold β. Then

$$P_{fa} = \int_{\beta}^{\infty} \ldots \int_{\beta}^{\infty} p_{N(\mathbf{s}_1,\ldots,\mathbf{s}_M)}(w_1,\ldots,w_M)dw_1 \ldots dw_M, \tag{9}$$

where $p_{N(\mathbf{s}_1,\ldots,\mathbf{s}_M)}(w_1,\ldots,w_M)$ is the joint M-variate probability density function for a background noise. Then, the joint probabilities of detection and missed detection are expressed in terms of the joint probability of the received signal, $p_{U(\mathbf{r};\mathbf{s}_1,\ldots,\mathbf{s}_M)}(w_1,\ldots,w_M)$:

$$P_d(\mathbf{r};\mathbf{s}_1,\ldots,\mathbf{s}_M) = \int_{\beta}^{\infty} \ldots \int_{\beta}^{\infty} p_{U(\mathbf{r};\mathbf{s}_1,\ldots,\mathbf{s}_M)}(w_1,\ldots,w_M)dw_1 \ldots dw_M, \tag{10}$$

$$P_{md}(\mathbf{r};\mathbf{s}_1,\ldots,\mathbf{s}_M) = \int_{-\infty}^{\beta} \ldots \int_{-\infty}^{\beta} p_{U(\mathbf{r};\mathbf{s}_1,\ldots,\mathbf{s}_M)}(w_1,\ldots,w_M)dw_1 \ldots dw_M. \tag{11}$$

The exact probability distribution functions for signal and noise may depend on signal modality and location of the measurements but, generally, can be described by a family of the chi-squared distributions. For high degrees of freedom, their first-order approximation would be the Gaussian distribution.

As an example, let us consider the effect of the dependency of two observations on the detection probability under assumptions of joint normal distributions for the signal and noise. Suppose the original (no noise) signal power at each of the two locations $\mathbf{s}_1$ and $\mathbf{s}_2$ has the mean $\mu_X = 4$ and vari-

ance $\sigma_X^2 = 2$, and noise is described by $\mu_N = 3$ and variance $\sigma_N^2 = 1$. The signal has been emitted at some location **r**. Assuming independence of the background noise and the signal, the distribution of the received (noisy) signal, $p_{U(\mathbf{r};\mathbf{s})}$, will be also normal with the mean $\mu_U = 7$ and variance $\sigma_U^2 = 3$. Suppose, the threshold for the false alarm is $P_{fa} = 10^{-4}$. Then, from equation (7), $\gamma = 6.7190$; and from equation (6), $P_d = 0.5644$. This is the individual probability of detection at each of the two locations.

If these observations were independent and the false alarm threshold was imposed to each of them individually, then equations (9–11) would recast into the products of univariate functions; and the joint probability of false alarm would be $P_{fa}(\mathbf{s}_1, \mathbf{s}_2) = 10^{-4} \cdot 10^{-4} = 10^{-8}$ whereas the joint probability of detection would be $P_d(\mathbf{r};\mathbf{s}_1, \mathbf{s}_2) = 0.5644 \cdot 0.5644 = 0.3186$. This result can be obtained directly in the probabilistic space using equations (4).

However, if these two observations were required to have jointly the false alarm probability of $P_{fa}(\mathbf{s}_1, \mathbf{s}_2) = 10^{-4}$, then, from equation (9), the detection threshold would be $\beta = 5.3263$ and the detection probabilities, using equation (10), would be $P_d(\mathbf{r};\mathbf{s}_1, \mathbf{s}_2) = 0.6940$ and $P_d(\mathbf{r};\mathbf{s}_1, \mathbf{s}_2) = 0.7544$, corresponding to completely independent and completely dependent observations, respectively. As one can see, in this example, applying the false alarm threshold to the entire set of observations significantly increased the detection probability for independent observations (more than two times). Furthermore, dependent observations increase the joint probability of detection in comparison with the independent ones (from 0.69 to 0.75). This corroborates with the previous statement that the assumption of independency represents the worst-case scenario for the joint probability of detection (Mission 2).

It is worth noting that this approach to calculating joint probabilities allows one to estimate the range of the possible probabilities for two limiting cases: for completely dependent and completely independent observations. Whatever true dependence is, the corresponding joint probability of detection lies in that range. For the considered example, partly dependent observations will result in the detection probability in the range [0.69 0.75].

3.1.3 Approximate metrics

Both missions considered in this study require joint probabilities, which can be easily calculated only for independent observations. However, this assumption is likely to be violated for spatially close route points. A more accurate approach of information fusion in the signal space can be employed, but it requires a specification of the joint probability density functions for signal and noise and multivariate integration in equations (9–11). For example, if a route is specified by $M = 50$ route points, then, 50-variate integrals should be evaluated numerically. It may be prohibitively complicated for some applications requiring a fast analysis. This poses a demand to have easily obtainable approximate metrics adequate for both dependent and independent observations. Reasonable choices are the maximal, $\max_{\{\mathbf{s}_{pq}\}} P_d(\mathbf{r};\mathbf{s}_{pq})$, and minimal, $\min_{\{\mathbf{s}_{pq}\}} P_d(\mathbf{r};\mathbf{s}_{pq})$, probabilities along the route for Mission 1 and Mission 2, respectively. For Mission 1, coverage provided with respect to the maximal instantaneous probability will hold true with respect to the actual probability $P_d^{(1)}(\mathbf{r})$ because the former is no greater than the latter. For Mission 2, coverage provided with respect to the minimal probability does not guarantee coverage in terms of the joint detection probability but guarantees that, at each individual route point, the probability of detection will not be smaller.

3.2 Probabilistic cost function

For missions requiring optimization of the aircraft detection (covert operations), the cost function in equation (1), **c**, should reflect the probability of the aircraft being heard at some specified locations, denoted as $\mathbf{r}_h$, where "listeners" are located. For this purpose, the aircraft plays the role of a sound source now rather than a moving sensor as we considered in the previous subsection. These locations are, in general, independent of the spatial points where coverage is required; they can be within the object being monitored or somewhere else.

Each element of cost vector **c** should reflect a probability of aircraft being detected along the entire route R_q. For a single individual waypoint, $\mathbf{s}_{\mathrm{pq}}$, it is reasonable to calculate the probability that at least one of the listeners will detect the aircraft. And the assumption of listeners' independence is justifiable in this case:

$$P_d^{(1)c}(\mathbf{s}_{pq}) = 1 - \prod_{j=1}^{J}\left(1 - P_d^c(\mathbf{r}_{hj};\mathbf{s}_{pq})\right), \tag{12}$$

where J is the number of listeners. To fuse probabilities for different $\mathbf{s}_{pq}$, we can apply the mathematical framework developed in the previous subsection with the only distinction being that aircraft is a sound source now and the sensors are the listeners. If the goal is to avoid the aircraft from being heard, a cost for a route R_q can be defined as maximal aircraft detection probability among all route points.

A most complicated but practically important case is when the listeners are located within (or close to) the object being monitored. This problem can hardly be solved without computer simulations for the quantitative assessment of both the aircraft coverage and aircraft exposure probabilities because the optimal route should balance these competitive demands. On the one hand, the aircraft should be close enough to the object of interest to provide necessary resolution of detection. But on the other hand, it should be far away from the listeners who may reveal and shoot it. Section 5 considers such a scenario. As the numerical simulations will show, the optimization algorithm with the probabilistic cost function, as defined above, will tend to choose a route or sensor locations farther away from the listeners (a safer operation). Sometimes, it will be preferable for aircraft to come closer to the object despite the higher risk of being exposed. This can be accommodated by introducing the risk tolerance factor, $c_0 \geq 0$; and the cost in equation (12) will be calculated as follows:

$$c_q = c_0 + \max_{\{\mathbf{s}_{pq}\}} P_d^{(1)c}(\mathbf{s}_{pq}). \tag{13}$$

The greater c_0 is, the higher the exposure risk that is accepted.

3.3 Coupling with ground sensors

The considered approach for aircraft route optimization can be used in practice in a few distinct ways. First, one can determine an optimal route (in terms of cost and provided coverage) for a single aircraft. Second, it is possible to determine an optimal combination of routes, numbers, and types of multiple aircraft. Third, the optimization can be performed to complement the coverage of a ground-sensor network. Finally, a complex problem of simultaneous optimization of ground sensors (numbers, types,

and locations) and aircraft routes can be solved. For the latter, the key mathematical quantities appearing in the original algorithm for stationary ground sensors (Vecherin et al. 2010, 2011), denoted with superscript S below, should be modified to include analogous quantities for aircraft, denoted with superscript UAS, as follows (a comma stands for merging matrices row-wise, and a semicolon stands for merging column-wise):

Coverage matrix: $\mathbf{A} \mapsto [\mathbf{A}^{S}, \mathbf{A}^{UAS}]$.
Indicator vector: $\mathbf{p} \mapsto [\mathbf{p}^{S}; \mathbf{p}^{UAS}]$.
Cost function: $\mathbf{c} \mapsto [\mathbf{c}^{S}, \mathbf{c}^{UAS}]$.

Note that the vector of coverage preferences, **b** in equation (1), remains unchanged. It sets coverage goals that will be achieved by determining an optimal combination (with respect to the total cost) of ground sensors and aircraft together.

4 Determining Candidate Aircraft Routes

In the framework described in Section 3, the candidate aircraft routes R_q should be specified, either manually or automatically, with flight planner software. In this section, we describe an algorithm to generate candidate routes for the missions outlined in Section 2. Generating a set of admissible routes is a key element in the set-covering route optimization paradigm. For our purposes, the following heuristics, assumptions, and constraints are employed to generate admissible routes:

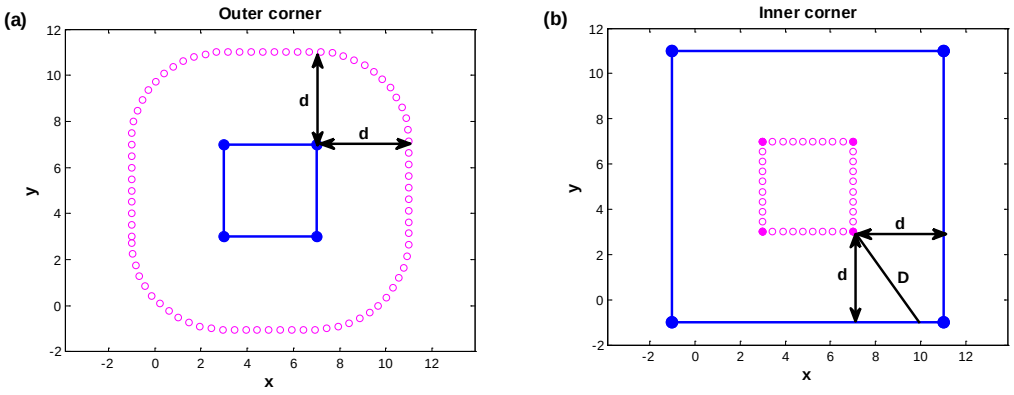
- If there are several areas of interest for a single mission, the areas are grouped into groups with no more than three areas of interest in each group.
- Optimization of multiple UASs (same or distinct types) is supported.
- The range of possible altitudes is specified by a mission planner.
- The spatial resolution is specified between the routes in vertical (Δz) and horizontal (Δr) directions; the minimal offset distances $d_{\min}^{def}$ and $d_{\max}^{def}$ may also be specified.
- The spatial resolution is specified between route points of a single route (Δs).
- The geometry of the area of interest determines the shape of the routes; that is, for each altitude, the route has a constant offset from the object except for problematic areas, as discussed below, where the offset can be larger.
- Optionally, irregular loitering routes with oscillations may be generated.
- Two or more UASs can follow the same route with delays.
- Detection from distinct UASs is independent.
- The areas of interest and the routes are specified by a set of point coordinates (a raster).
- The airspeed of a UAS and its endurance are specified.
- Optionally, the sensor's field of view, a preferred tilt angle, and required target resolution can be specified.
- For each altitude, the admissible offsets will not exceed the UAS endurance and, if specified, will be compatible with the required sensor's field of view and resolution.
- No-fly zones are supported.

4.1 Basic algorithm for candidate route generation

While constant altitudes, both above the sea level and above the ground level, are trivial to handle numerically, constant offsets are not, especially when the objects of interest have irregular shapes.

As Figure 2(a) demonstrates, for convex objects, the constant offset requirement is realizable for the exterior space, where d denotes the offset distance. A square in the center is a convex object subject to monitoring, and the purple curve around it is an aircraft route distant from each point of the object at the specified distance d. However, as Figure 2(b) illustrates, the problem is infeasible for the interior space. Apparently, a geometric figure with a constant offset from the edges of the external square (representing an object of interest) would be another square in the interior (representing aircraft route). However, the requirement to have a constant offset will be violated for all points of the external object closer to the lower right corner, as illustrated in Figure 2(b), because $D > d$.

Figure 2. Aircraft route (purple circles) with fixed offset from a convex object of interest (blue square): (a) outer space and (b) inner space. Note that the condition of the exact offset distance d is violated at sharp angles (the lower-right corner) because $D > d$.

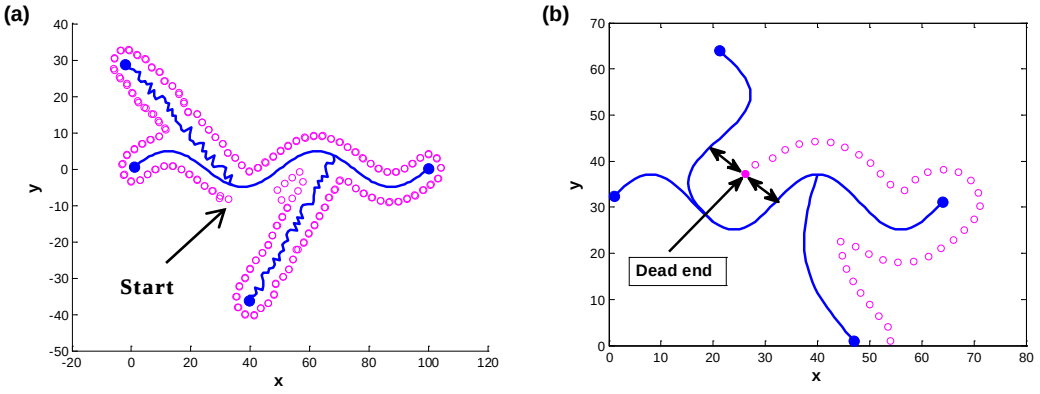


For realistic scenarios (e.g., roadways of irregular shapes), sharp corners are frequently present; and aircraft may face a problem encircling both the outer and inner corner space. To make the problem geometrically feasible, the requirement of having an exact offset distance should be relaxed to "not closer than the specified distance but as close as possible." Even after such a relaxation, it is still challenging to find a robust and efficient numerical solution; we discuss some of the numerical issues below.

Self-intersections represent a common routing problem as illustrated in Figure 3(a). In this figure, the blue curve depicts a roadway, and the purple

circles depict a UAS route at some specified offset distance (with the exception of sharp corners where the offset is allowed to be larger). The route starts at the circle with coordinates (35, −7) and goes clockwise until the problematic sharp-angle area. There, once the self-intersection happens, it goes backwards toward the beginning because previous route points are, indeed, the most optimal ones. To avoid such an issue, the condition forbidding self-intersections with all previously placed route points should be imposed, which may notably degrade the speed of the algorithm, especially, for long routes.

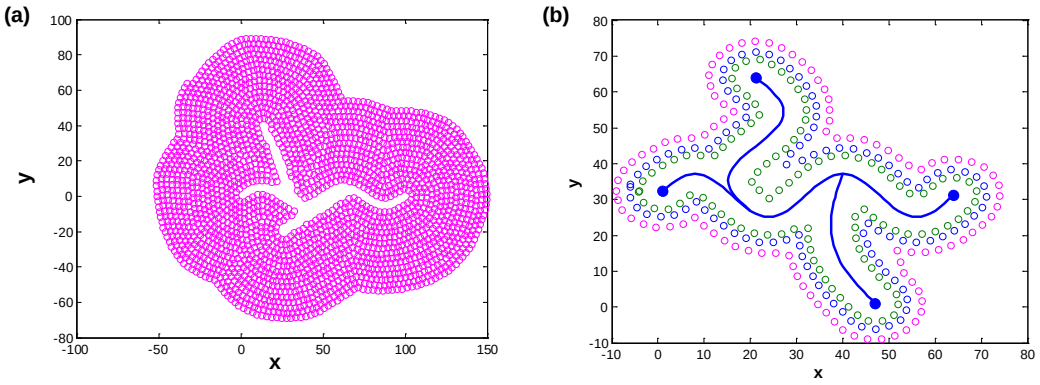
Figure 3. Illustration of failures in automatic route generation: (a) self-intersecting routes, and (b) bottleneck resulting in abrupt route termination.



A less apparent issue is a bottleneck that leads to an abrupt route termination, as illustrated in Figure 3(b). It can happen when there is a very narrow entrance satisfying the requirement of the specified offset distance, so that an exit is possible only by intersecting previous route points. Because such intersections are forbidden, the algorithm terminates. The only way to resolve the issue is to prevent the search algorithm from entering the bottle neck. To this end, the algorithmic solution is to go back one step, temporarily increase the offset distance and the distance between the route points, and repeat the search of the new route point with these new parameters. If the first iteration did not resolve the issue, increase the distances again and repeat the search. Eventually, the algorithm will avoid entering the problematic area. At this moment, the distances are set back to their original values, and the search continues in a normal mode.

With these counter-measures introduced in the algorithm, a new problem of infinite looping may appear, as shown in Figure 4(a).

Figure 4. Illustration of (a) the infinite loop failure and (b) a final algorithm with all modifications. Note routing in the problematic sharp-angle areas.



The reason for infinite looping is that the termination criterion for the search is the proximity of the last route point to the first one. However, if the starting point was accidentally selected in a problematic area, no route point will fall in the starting point's close proximity, and the algorithm will continue to search. To resolve this issue, the starting point should be selected at the most remote pixel of the object, such that the route converges to the same route point from both directions.

With these modifications, the final algorithm for automated generation of admissible routes reads as follows:

Algorithm 1. Basic routing.

Inputs: (x0, y0) – Cartesian coordinates of the object of interest; d0 – minimal offset distance; ds – minimal distance between two consecutive route points; dx0 – map resolution.

Output: (x, y) – Cartesian coordinates of the route.

```
1   Select the first route point;
2   Set distance from the current point to the first point: D = ds+1;
3   WHILE (D > ds)
4       FOR each candidate route point
5           set d2 = Inf; the closest distance to the object;
6           make a new candidate route point at the distance ds from the
7           last one;
8           calculate the minimal distance d from the current point to all
9           object's points;
```

```
10          calculate the minimal distance d1 from the current point to all
11          previous points;
12          IF (d < d2) AND (d >= d0) AND (d1 >= ds)
13              d2 = d;
14              select the current candidate;
15          END IF
16      END FOR
17      IF (no candidate selected)
18          update a list of problematic candidates;
19          increase d0;
20      ELSE
21          record the selected candidate as a new route point (x, y);
22          set d0 to its original value;
23          calculate D;
24      END IF
25  END WHILE
```

The described algorithm has been tested for various objects of regular and irregular shapes and has performed reliably. For the square object, depicted in Figure 2(a) as a blue square, the algorithm provided the known theoretical solution for outer space, depicted as a purple curve in Figure 2(a). Figure 2(b) depicts a more complex example of a highly irregular roadway with four problematic sharp-angle areas. As one can see, the algorithm works equally well in this case.

4.2 Introducing various complications

In practice, one may desire incorporating some specific constraints into the problem. We consider a few of them below.

4.2.1 Oscillatory loitering patterns

There two main reasons for having oscillatory routes in the admissible set. First, coverage may be required in a certain area; but an exact location of the target is unknown. In this case, loitering somewhat around the main course of the route might be helpful. Second, there might be tactical considerations, such as to make aircraft movement less predictable or to simulate its leaving the scene.

To achieve this, two pivot routes (of the same length in terms of route point) need to be specified, R_o and R_1, corresponding to the closest and the farthest offset distances, respectively. R_1 is determined using the basic routing algorithm. R_0 is determined using R_1. (In principle, it could have been determined, also, using the basic algorithm; but, in this case, the number of route points would be different in R_0 and R_1.) First, R_1 is described in the polar coordinate system relative to an arbitrary origin somewhere inside R_1 (e.g., taking the mean values of the R_1 Cartesian coordinates x and, separately, y). Second, the magnitudes of the radius vectors are reduced to match the specified closest offset distance. Third, R_0 route points are determined for the same polar angles but using the adjusted magnitudes of the radius-vectors. Finally, the x- and y-coordinates of the oscillatory route points are found by using the following equations:

$$\begin{aligned} x_p &= 0.5(x_{1p} + x_{0p}) + 0.5(x_{1p} - x_{0p})\cos\left(2\pi F(p-1)/M\right), \\ y_p &= 0.5(y_{1p} + y_{0p}) + 0.5(y_{1p} - y_{0p})\cos\left(2\pi F(p-1)/M\right), \end{aligned} \tag{14}$$

where (x_p, y_p) are the p-th coordinates of the oscillatory loitering route, $p = 1,...,M$; (x_p, y_p) and (x_p, y_p) are coordinates of the closest and farthest routes, respectively; and F indicates oscillation frequency. Figure 5 shows oscillatory routes for oscillation frequencies F = 1, 2, 6, and 8. The pseudocode for the loitering algorithm is given below:

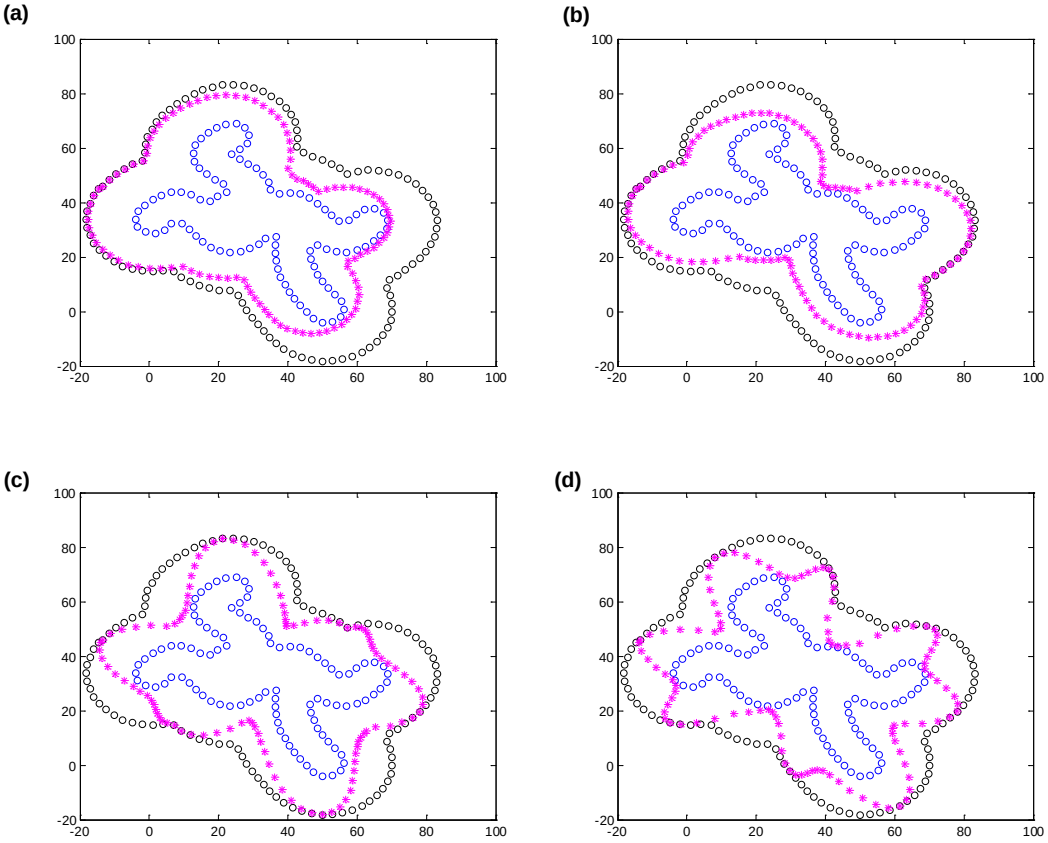
Algorithm 2. Oscillatory loitering routes

Inputs: (x1, y1) – Cartesian coordinates of the farthest route R1; d – reduction distance; F – oscillation frequency.

Output: (xp, yp) – Cartesian coordinates of the oscillatory loitering route.

```
1   select the polar coordinate system origin;
2   calculate radius vectors r and polar angles for each route point of R1;
3   reduce the radius vectors on the specified reduction distance: r=r-d;
4   determine route R0 using the adjusted r and the same polar angles;
5   FOR each candidate route point
6       use Eq. (14) to determine the loitering route;
7   END FOR
```

Figure 5. Oscillatory loitering routes for different frequencies. Blue and black markers indicate the closest and the farthest routes, R_0 and R_1, respectively. Magenta markers depict the oscillatory route. *F* is the oscillatory frequency indicating how many times the outer route will be touched by the oscillatory route: (a) $F = 1$, (b) $F = 2$, (c) $F = 6$, and (d) $F = 8$.



4.2.2 No-fly zones

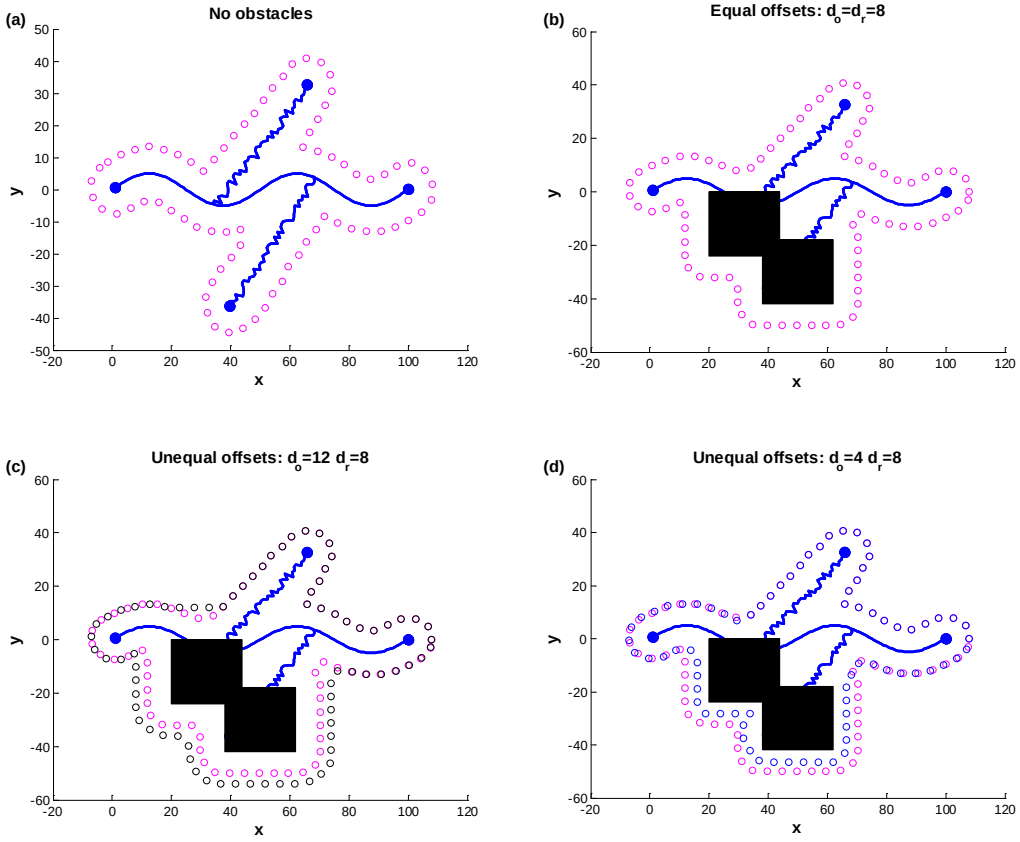
No-fly zones are a common requirement for automated aircraft routing. These zones can represent civilian residential areas, places of religious importance, or can be forbidden for tactical considerations. These requirements can be handled in optimization either as *soft* constraints or as *hard* constraints. The soft constraints penalize the objective function for any part of a route passing over the no-fly zones. However, the penalty should be chosen very carefully as arbitrarily large values will make the objective function non-differentiable (or poorly differentiable), which is a principal violation in some optimization approaches, jeopardizing the operability of the entire optimization, whereas not-so-large values will allow some short segments of a route to pass over the no-fly zones. The optimization approach described in this report allows one to impose the hard constraints, that is, routes (or any part of theirs) are guaranteed to not pass over specified no-fly zones.

If the offset distance from a no-fly object coincides with the offset distance from the object of interest, the no-fly object can simply be treated as another object of interest. However, in reality, the desired offset distances from the object being monitored and the no-fly object might be distinct. This requires a modification of the basic routing algorithm. First, a new input variable, d_{obs}, is added to indicate the desirable offset distance from the obstacle. Second, the selection of the first route point should be adjusted to be not closer than d_{obs} from the no-fly zone. Third, for each new candidate route point, the closest distance to the no-fly area must be calculated, D_{obs}. And fourth, the selection criteria of a new route point, lines 12 and 13 in Algorithm 1, should be modified as follows:

12 IF (d_{min} < d2) AND (d >= d0) AND (d1 >= ds) AND (D_{obs} >= d_{obs}),
13 d2 = d_{min},

where d_{min} = min(d, D_{obs}). Figure 6 illustrates the operability of the modified algorithm for equal and unequal offset distances. Figure 6(a) depicts the route without any no-fly zones. Figure 6(b) shows how the original route would change in the presence of the no-fly zone depicted by black squares and with the same offset distances from both the roadway and the obstacle. Figure 6(c) depicts the routing around the no-fly zone when the offset distance from the zone is larger than the offset distance from the roadway (the black markers). In the areas far enough from the no-fly zone, the route coincides with the original route (without the no-fly zone), while the routing is distinct in the vicinity of the zone. Especially, note the behavior in the upper “valley” where the required offset from the no-fly zone dominates the requirement of the roadway offset. Figure 6(d) depicts the opposite situation when the no-fly zone offset is smaller than the roadway offset. These figures illustrate that the algorithm yields reasonable solutions in the presence of no-fly zones.

Figure 6. Avoiding no-fly zones (black squares): (a) route without no-fly zones, (b) roadway and no-fly zones with equal offset distances, (c) offset distance from the no-fly zone is larger than from the roadway (the magenta markers indicate the route with equal offset distances and the black markers depict the unequal offset route), and (d) offset distance from the no-fly zone is smaller than from the roadway (the magenta markers indicate the route with equal offset distances and the blue markers depict the route with an unequal offset).



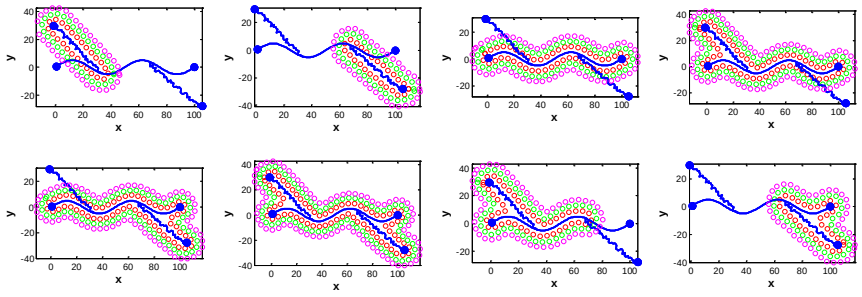
4.2.3 Multiple targets

It is not uncommon to need to observe multiple objects of interest during a single UAS flight. These can be disconnected objects or branches of a single long roadway. Each of the objects may have its own unique geometry and, thus, will generate a set of its own admissible routes. Furthermore, a route can be generated embracing all of them at once. Additionally, candidate routes can be generated around each possible combination of the specified objects or only around some specific combinations. With the first option, the number of objects should be limited to avoid a rapidly increasing number of possible combinations. For example, if the number of objects is N_{obj} = 3, numbered 1, 2, and 3, then for each allowed altitude and each offset distance, there are 3 routes around each object, one route around all of them together, {1, 2, 3}, and 3 routes around distinct combi-

nations, such as {1, 2}, {1, 3}, and {2, 3}. Altogether, there are 7 candidate routes. Suppose four possible altitudes and three possible offset distances are specified. Then, 3 objects will generate 84 candidate routes. Additionally, the oscillatory loitering routes can be included for each altitude. As one can see, the set of admissible routes is rapidly increasing with the number of objects when the combinatorial option is used. Each of these routes is characterized by a set of M route points; M is determined by a specified resolution and the size of the object (in practice, M can be several tens up to a maximal number allowed by a specific UAS control unit). In the current program realization, the number of objects is limited to 3 to keep the calculations tractable for an average personal computer. If there are more objects of interest for a single mission, we recommend either using partial combinations or grouping these objects in clusters of three and generating routes for each cluster individually.

Figure 7 depicts the combinatorial routes around three branches of a roadway (the blue curve). Markers of different colors correspond to different offset distances.

Figure 7. Candidate routes around branches of a long roadway. Markers of different colors correspond to different offset distances.



4.2.4 Camera's field of view and resolution

Some missions may have specific requirements for the quality of images being acquired, such as the resolution and a view angle of a target. For example, Etemad and Chellappa (1997) used the image resolution of 50 × 60 pixels for automated face recognition program FERET (Phillips et al. 2000) and showed that this resolution can be reduced under some circumstances. Other image recognition algorithms may have different requirements. Johnson's criteria (Johnson 1958) set minimal resolution requirements for different levels of target recognition by a human. Although the original criteria were specified in terms of line pairs on the target image, suitable for CRT (cathode ray tube) monitors, they can be recast into pix-

els, suitable for LCD (liquid crystal display) monitors and digital imaging, assuming that each line pair corresponds to 2 pixels, as follows:

- Detection of target's presence: 2.0 ± 0.5 px
- Detection of target's orientation and symmetry: 2.8 ± 0.7 px
- Detection of the type of target (a person versus a car): 8 ± 2 px
- Identification of the target (a woman versus a man, a specific car): 13 ± 3 px;

These criteria correspond to a 50% probability to detect a target at a desired level. For a reliable detection, the target resolution should be greater.

A preferred view angle, which determines the tilt of the camera's optic axis relative to the vertical, may be required for several reasons, such as mitigating sunlight reflection and target recognition (e.g., one can hardly distinguish a car's model observing it from above at a right angle).

The algorithm for generating admissible routes requires specification of a flight altitude and an offset distance. However, these values are not arbitrary. The altitude can be selected from the allowed range (determined by a mission planner), $[h_{\min}^{def}, h_{\max}^{def}]$, with the specified altitude increment Δz. Here, $h_{\min}^{def}$ and $h_{\min}^{def}$ are the minimal and maximal flight altitudes set in advance by a mission planner for each type of UAS (the superscript *def* stands for *default*). For each altitude, the offset distance d varies from some $d_{\min}^{def}$ to some $d_{\max}^{def}$, also specified by a mission planner. If the preferred offset distance range is unspecified, $d_{\min}^{def}$ equals the increment of the offset distances Δr whereas $d_{\max}^{def}$ corresponds to the longest possible route the aircraft is capable to fly, which depends on its speed (and, thus, fuel consumption), maximal time in flight (endurance), and the size and shape of the area of interest. For example, $d_{\max}^{def}$ can be determined requiring the flight time not to exceed the UAS endurance. The Appendix presents the calculation of the flight time along a route with and without a four-dimensional wind field. These default constraints on the altitudes and offset distances may be refined to take into account required specifications of image quality and appearance, as considered below.

There are many factors affecting the resolution of the image, such as quality of the optics, the lighting conditions, and the pixel resolution. The latter is considered in this report. Suppose, a target of a characteristic size l

should be observed with the resolution equal to or greater than E pixels. Then, the image size, l_I, on the camera's sensing element should be equal to or greater than Ew, where w is the size of a single pixel in the camera's sensing element which is determined by camera's technical specifications. The image size can be determined from the following equation (Abrams et al. 1999) valid in the three-dimensional space:

$$l_I = \frac{sl\left\|\left[\left(\mathbf{r}_a - \mathbf{r}_v\right)\times\mathbf{u}\right]\times\mathbf{v}\right|}{\left[\left(\mathbf{r}_a - \mathbf{r}_v\right)\cdot\mathbf{v}\right]\left[\left(\mathbf{r}_b - \mathbf{r}_v\right)\cdot\mathbf{v}\right]}, \tag{15}$$

where

s = the distance from the back nodal point of the lens to the image plane,
$\mathbf{r}_a$ and $\mathbf{r}_b$ = the radius vectors of the endpoints of the target of size l,
$\mathbf{u}$ = the unit vectors from $\mathbf{r}_a$ to $\mathbf{r}_b$,
$\mathbf{r}_v$ = the radius-vector of the lens' front nodal point,
$\mathbf{v}$ = the unit vector along the optical axis.

For the considered case of a camera mounted on a UAS, equation (15) can be simplified. Because the distances from the camera to the target are much larger than the target size, $\left(\mathbf{r}_a - \mathbf{r}_v\right) \approx \left(\mathbf{r}_b - \mathbf{r}_v\right) = D\mathbf{v}$, where D is the distance from the UAS to the target. Considering a cross-section of the target on the ground in the vicinity of the optical axis, $\mathbf{u}$ is perpendicular to $\mathbf{v}$. Finally, taking into account that the camera is focused to infinity, $s = f$, where f is the focal length of the lens. With these adjustments, the requirement of minimal image resolution can be recast into the estimation of the maximal distance from the UAS to the target, $D_{\max}$, as follows:

$$D_{\max} = \frac{lf}{Ew}, \tag{16}$$

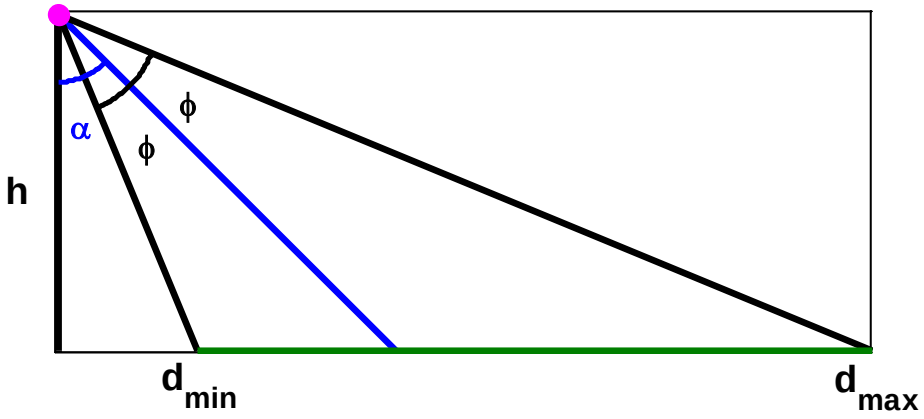
which can be used to determine both the maximal altitude above the ground, $h_{\max}^{res}$, and, for each smaller altitude, h, the maximal offset distance, $d_{\max}^{res}$.

$$h_{\max}^{res} = \sqrt{D_{\max}^2 - \left(d_{\min}^{def}\right)^2},$$
$$d_{\max}^{res} = \sqrt{D_{\max}^2 - h^2}, \quad h_{\min}^{def} \le h \le h_{\max}^{res}. \tag{17}$$

A mission planner can specify a preferred view angle, α, by indicating the tilt of the camera's optic axis relative to the vertical, as shown in Figure 8. Suppose, the camera's field of view in the vertical direction is characterized by angle $\varphi > 0$. Then, as can be seen in Figure 8, for a given h, the offset distances should be in the range $[d_{\min}^{fov}, d_{\max}^{fov}]$:

$$d_{\min}^{fov} = h\tan(\alpha - \varphi), \quad d_{\max}^{fov} = h\tan(\alpha + \varphi). \tag{18}$$

Figure 8. Geometry of the camera's vertical field of view. The magenta marker indicates the UAS. The blue line indicates the camera's optic axis deviated from the vertical on the specified angle α. The camera's field of view is centered on the optical axis in the range [α − φ, α + φ]. A target must be within [d_{min}, d_{max}] range (green line) to be within the camera's field of view.



Equations (18) can be used to determine the minimal and maximal altitudes above the ground, $h_{\min}^{fov}$ and $h_{\max}^{fov}$, respectively, for a specified tilt of the optical axis and camera's field of view:

$$h_{\min}^{fov} = \frac{d_{\min}^{def}}{\tan(\alpha + \varphi)}, \quad \alpha \ge 0, \alpha + \varphi < \frac{\pi}{2},$$
$$h_{\max}^{fov} = \frac{d_{\max}^{def}}{\tan(\alpha - \varphi)}, \quad \alpha > \varphi. \tag{19}$$

Another optical characteristic that may affect image quality is the camera's depth of view (Abrams et al. 1999), which depends on the camera's aperture. The latter, in its turn, depends on lighting conditions and on the light

sensitivity of the camera's sensing element, which may be electronically enhanced. These parameters are automatically set, and thus, for the considered UAS application, we assume that the depth of view is not a concern.

Thus, given default ranges of possible altitudes, $[h_{\min}^{def}, h_{\max}^{def}]$, and offset distances, $[d_{\min}^{def}, d_{\max}^{def}]$, the operational ranges can be found using the following formulas:

$$\begin{aligned} h_{\min} &= \max\left\{h_{\min}^{def}, h_{\min}^{fov}\right\}, \quad h_{\max} = \min\left\{h_{\max}^{def}, h_{\max}^{fov}, h_{\max}^{res}\right\}, \\ d_{\min} &= \max\left\{d_{\min}^{def}, d_{\min}^{fov}\right\}, \quad d_{\max} = \min\left\{d_{\max}^{def}, d_{\max}^{fov}, d_{\max}^{res}\right\}, \quad h \in [h_{\min}, h_{\max}]. \end{aligned} \tag{20}$$

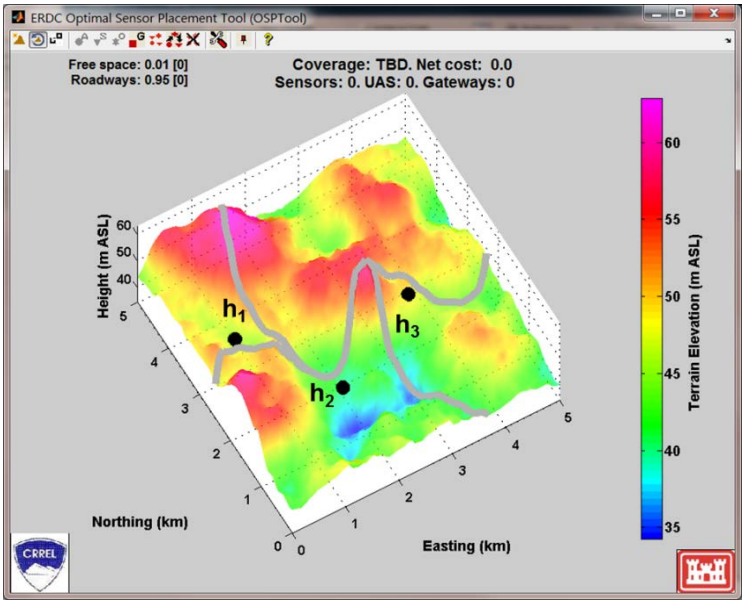
Note that the estimates of the operational ranges in equations (20) depend on the camera's focal length, f, and the vertical field of view, φ. For cameras with zoom, these two quantities are dependent, $\varphi = \varphi(f)$ and vary in certain ranges. For larger focal lengths, one can increase the maximal detection distance, in accordance with equation (16), at the expense of the camera's field of view. For such cameras, the range estimates should be implemented for every focal length that will be used during the UAS's operation, which will generate additional sets of candidate routes.

5 Exemplary Scenario: Monitoring a Roadway in Mountainous Terrain

This section considers the following scenario. A friendly troop is going to take a route to its destination. Along the route, on the roadsides, enemies may lie in wait at certain likely locations. These places are not seen from the road. A commander decides to use UAS to make sure that the route is secure. The UAS should provide the best resolution imagery (i.e., to fly as close to the roadsides as possible) but, at the same time, avoid being heard and shot (minimizing its audible footprint at the hostile locations).

Figure 9 illustrates this scenario. A gray roadway is the object of interest. The color of the terrain reflects terrain elevation above sea level. Three black circles at the locations $\mathbf{h}_1$, $\mathbf{h}_2$, and $\mathbf{h}_3$ represent hostile observers along the roadway. The goal of this section to test and verify the operability of the optimization approach described in this report. The focus of the numerical example is not to accurately model a specific UAS equipped with a specific payload but to investigate what the solution would be for given coverage and cost. For this purpose, we assume that a camera is omnidirectional and that the actual flight altitudes for different UAS types are irrelevant.

Figure 9. An example of the surveillance scenario. A gray roadway is the object of interest. Black circles depict locations of hostile observers.



The admissible UAS routes are generated as described in Section 4. Altogether, 112 routes are generated. Figure 10 shows some of the routes in which the vertical axis is scaled to show the terrain effects. Figures 10(a) and (b) depict routes at three allowed altitudes above the sea level at three offset distances (black solid curves). Figure 10(c) shows routes at a constant altitude above the ground. Figure 10(d) exemplifies an oscillatory loitering route with the oscillation frequency $F = 8$.

Figure 10. Examples of admissible routes: (a) and (b) routes with constant offset distances and constant altitudes above the sea level, (c) routes with constant altitudes above the ground, and (d) an oscillatory loitering route.

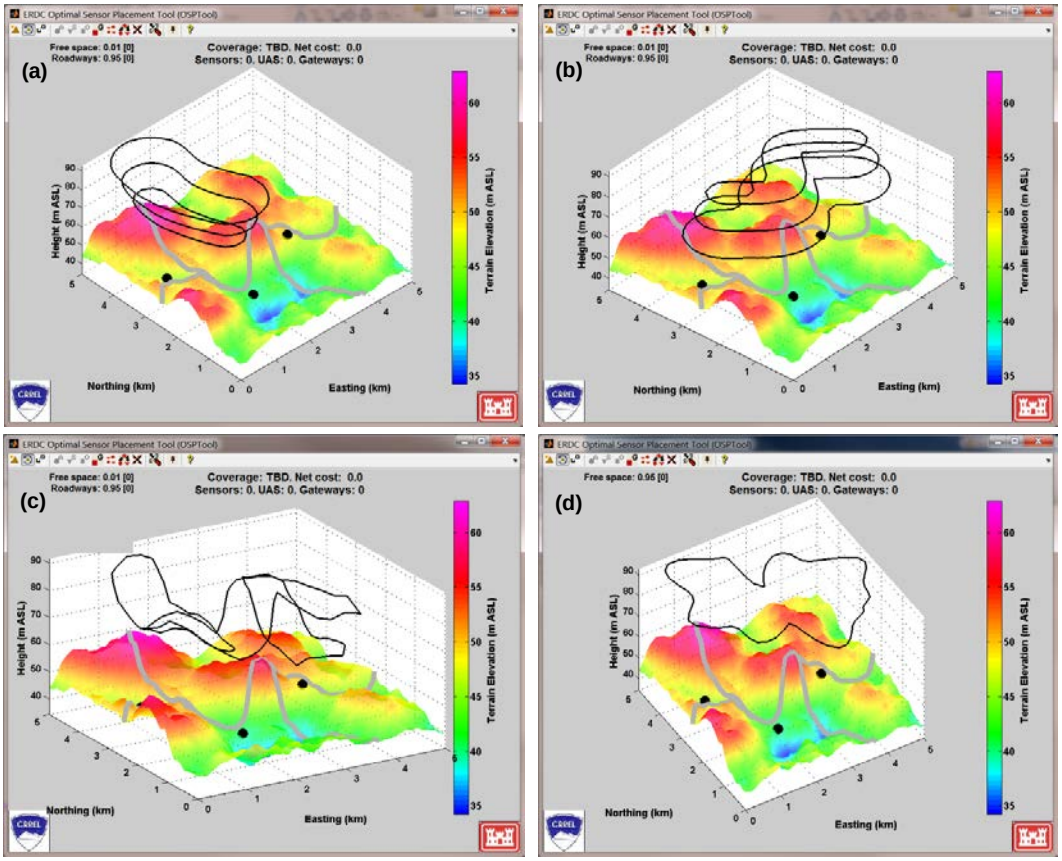
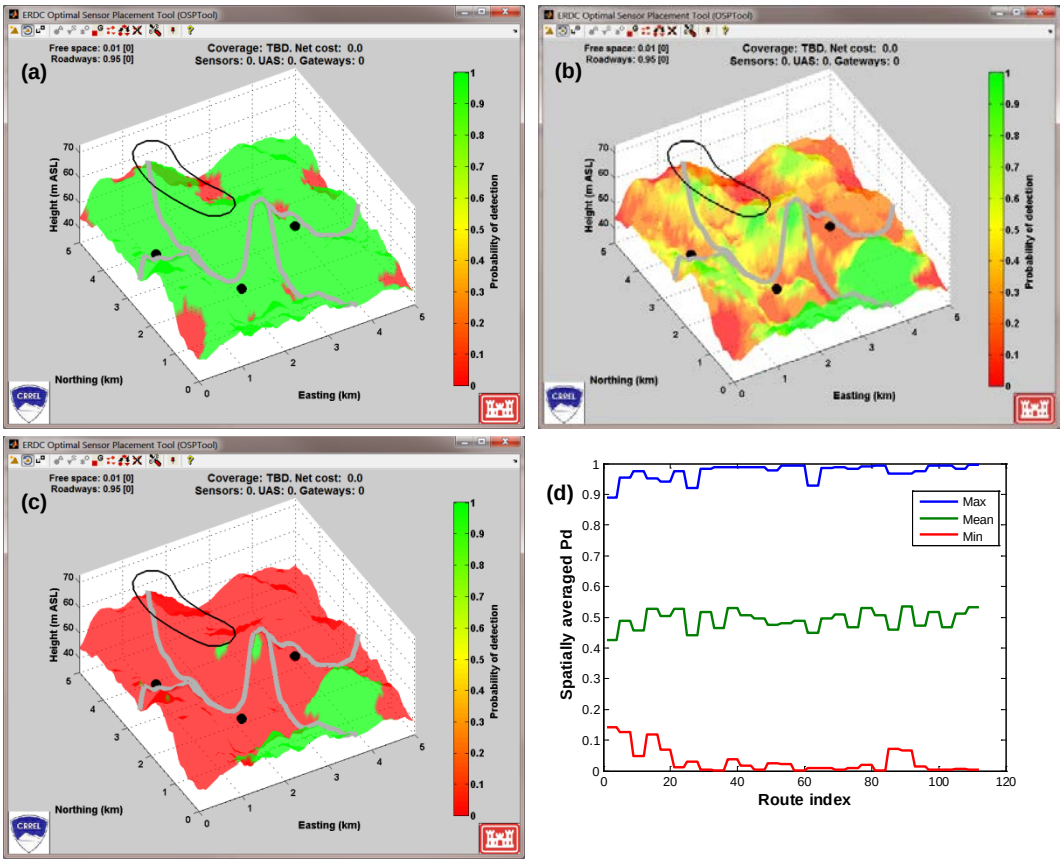


Figure 11 shows an example of UAS coverage for one admissible route; the color of the terrain represents the probability of detecting an event (enemy presence, a target, etc.) on the ground. Figures 11(a), 11(b), and 11(c) show the maximal probability of detection for each pixel on the ground, the average probability along the route, and the minimal probability of detection, respectively. Green and red colors mean high and low probabilities of detection, respectively. As one can see, different locations on the ground are covered with different probabilities. To characterize an entire route at-

a-glance, Figure 11(d) shows the spatially averaged probability of detection for each of the admissible routes.

Figure 11. UAS coverage characterized by different metrics: (a) each pixel on the ground is colored to represent the maximal probability of detection along the UAS route, (b) route-averaged coverage, (c) minimal probability of detection, (d) spatially averaged probability of detection for all admissible routes.



The outlined scenario task of confirming or denying the presence of a target fits Mission 1. Therefore, an appropriate metric for "goodness" of a UAS route is the maximal probability of detection. Let us consider several tactical cases.

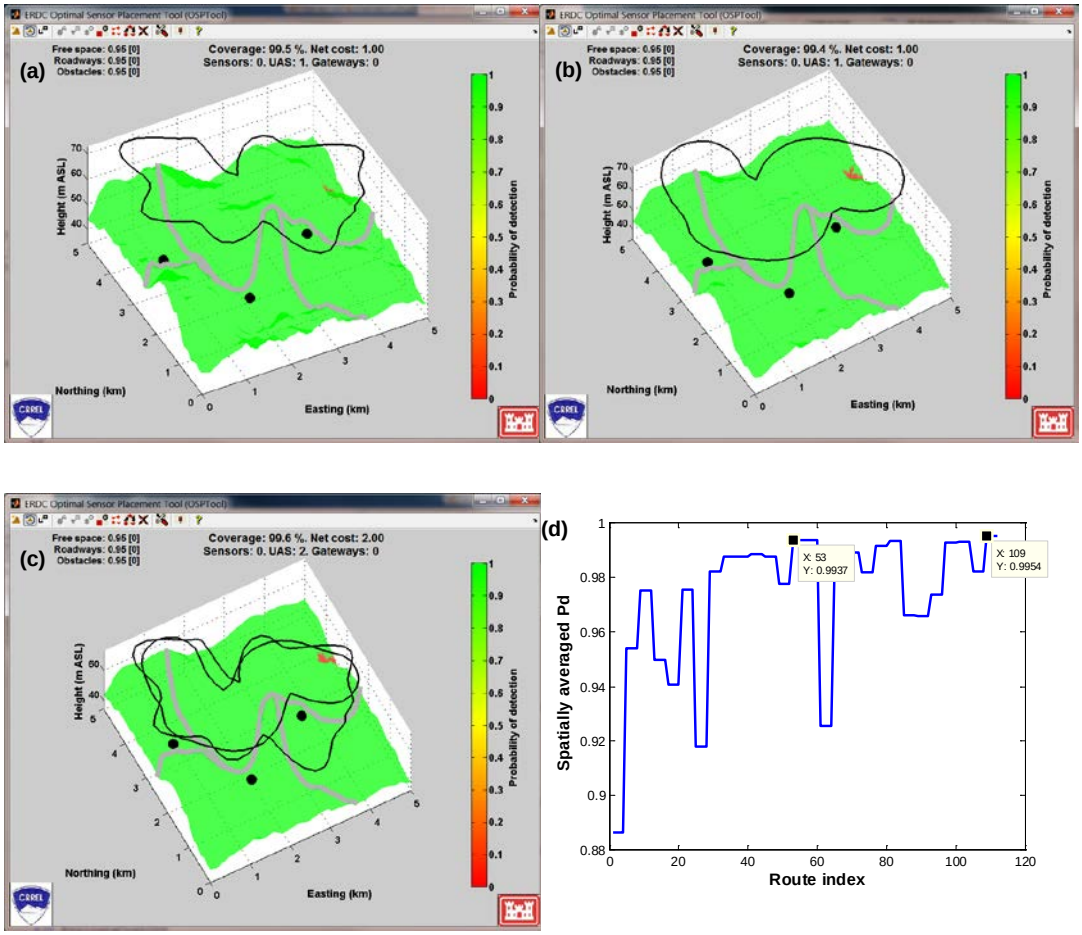
Case 1:

- All routes are above the audibility threshold, that is, a UAS is not heard on the ground for any route it is flying on.
- Coverage preferences are 0.95 or greater for each pixel on the scene, which includes both the roadway and the surrounding space.
- No ground sensors are available.
- The number of UASs is limited to 1 (or 2) to manually verify the solution.

For this case, the cost equals one for every route, and the problem reduces to minimization of the number of UASs. Because the coverage preferences are equal for each ground location and the number of UASs is limited to 1, the solution to this problem can be easily determined analytically: it should be a single route with the widest possible coverage. The latter can be inferred from Figure 11(d), blue curve.

Let us verify whether this solution is suggested by the algorithm. The numerical solution is route 109, which is an oscillatory loitering route at the lowest altitude, shown in Figure 12(a). If the loitering routes were not present in the admissible set, the optimal solution would be route 53, which is the route with the lowest altitude and largest offset distance (Figure 12[b]). The corresponding spatially averaged coverage for these routes is 0.9954 and 0.9937, respectively, as seen in Figure 11(d) and, in greater details, in Figure 12(d). That is, the numerical solutions coincide with the predicted analytical ones. If the number of UAS were limited to 2, the numerical solution is depicted in Figure 12(c). In this case, it consists of two routes with the widest individual coverage. In general, this is not the case because the widest individual coverage footprints do not automatically result in the widest coverage together due to overlapping coverage. Note that coverage preferences of 0.95 detection probability were not satisfied for all locations on the ground. In the vicinity of the northeastern corner, there is a terrain nook with poor coverage indicated in red.

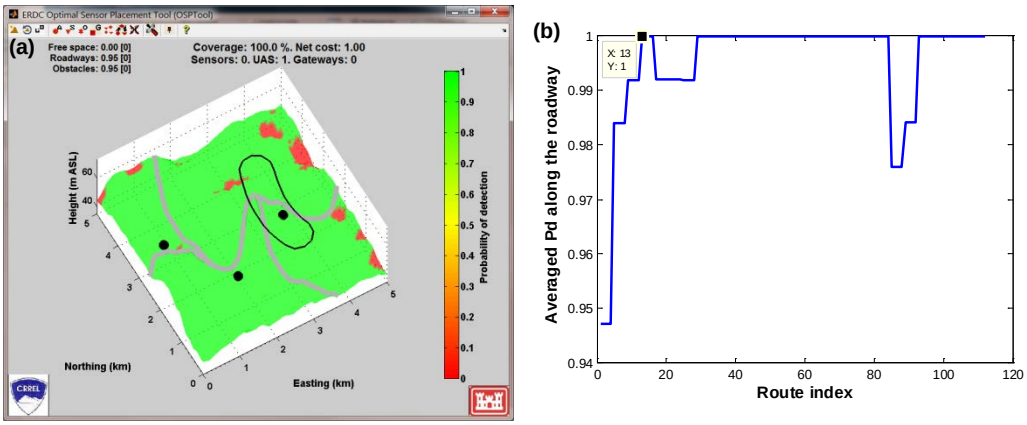
Figure 12. Numerical solutions to Case 1: (a) the optimal route (109) when the oscillatory loitering routes are included in the set of candidate routes; (b) the optimal route (53) when the loitering routes are excluded from the set of candidate routes; (c) the optimal combination of routes (109 and 53) when the number of UAS is limited to 2; and (d) the spatially averaged probability of detection, verifying the numerical solution.



Case 2: the same as Case 1, but coverage is required for the roadway only, $P_d^{pref}(\text{roadway}) = 0.95$ and $P_d^{pref}(\text{surrounding space}) = 0$.

As one can see in Figure 13(a), the optimal route (13) for Case 2 is distinct from the Case 1 solution. It provides the required coverage for every spatial location with just a single UAS. Figure 13(b) shows the probability of detection averaged over the roadway and verifies the correctness of the suggested solution.

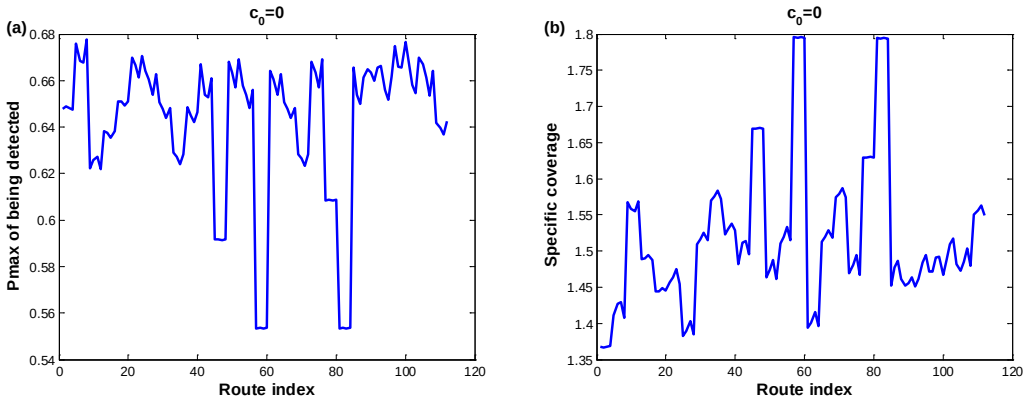
Figure 13. Optimal routes for Case 2: (a) the optimal route (13) and (b) the spatially averaged probability of detection over the roadway, verifying the correctness of the numerical solution.



Case 3: the same as Case 1, but each route is being heard on the ground and characterized by the probability of being detected, as indicated by equations (12) and (13).

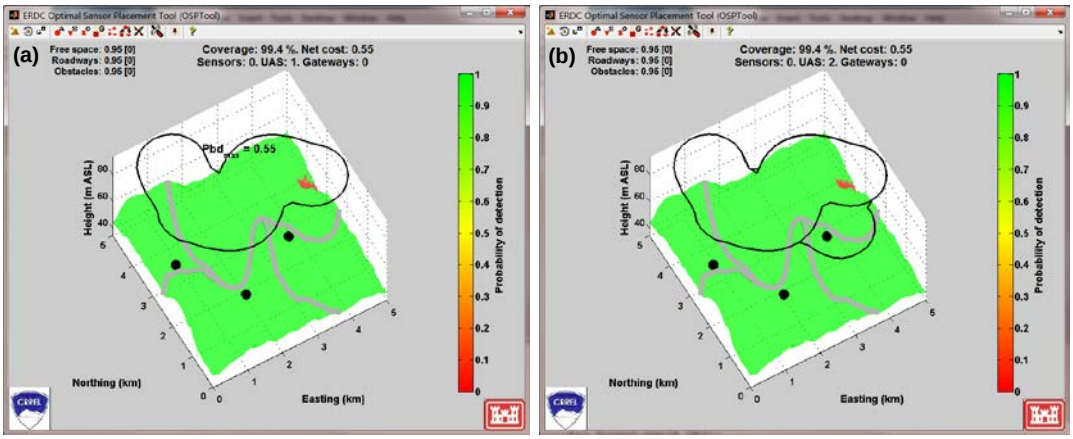
Let us consider first when the risk tolerance factor is set to zero, $c_0 = 0$. Then, the cost reflects the maximal probability along the route that at least one of the listeners will detect the UAS (Figure 14[a]). The algorithm determines an optimal route that balances these two competing factors, coverage and cost. The quality of the solution can be inferred from the specific coverage, which is coverage per unit cost, shown in Figure 14(b). In general, optimal solutions provide higher specific coverage.

Figure 14. Cost function with the risk tolerance factor $c_0 = 0$: (a) maximal probability of being detected by at least one listener and (b) specific coverage per unit cost. In general, optimal routes correspond to high values of the specific coverage.



Figures 15(a) and (b) show numerical solutions derived by the optimization algorithm when the number of available UASs is limited to 1 and 2, respectively.

Figure 15. Optimal routes for Case 3: (a) optimal route (59) when the number of UASs is limited to 1 and (b) optimal routes (59 and 83) when the number of UASs is limited to 2. These routes correspond to two peaks of specific coverage seen in Figure 14(b).



There are three features worth noting. First, the suggested routes (59 and 83) correspond to the highest specific coverage, two peaks seen in Figure 14(d), as expected. Second, the single route in Figure 15(a), 59, is almost identical to previously suggested route 53, Figure 12(b), except that route 59 is at the maximal altitude whereas route 53 is at the lowest altitude. This conforms to intuitive anticipation that UAS should fly as high as possible to minimize its audio footprint. Third, the two routes shown in Figure 15(b) are almost identical. In fact, for some scenarios, the most optimal solution might be to have two (or more) UASs fly the same route. Such a situation is not uncommon in guarding practice. If there is a perfect route to guard an object but the time interval of a single guard to make a complete loop is too long, the practical solution would be to employ another guard following exactly same route with a delay. Thus, if multiple UASs are suggested to fly the same route, it can be assumed that their relative delays are equal and evenly distributed along the route, such that two UASs are separated by the half of the route, three UASs by one third, etc.

The effect of the distinct risk tolerance factors in the cost, c_0, is better understood using optimization of ground sensors because the underlying tendency appears more distinctly. Suppose four ground sensors were supplied, two acoustic (small black bullets in Figure 16) and two seismic (white triangles), to cover the same scene under the conditions of Case 2, that is, the objective is to cover the roadway only while minimizing the probability for friendly sensors of being detected by the three hostile observers. The risk tolerance factor varies in the range [0, 1.75] with an increment of 0.25. Figure 16(a), (b), (c), and (d) show optimal locations of the ground sensors for a few selected values of the risk tolerance factor c_0 =

0, 0.5, 1, and 1.5, respectively. Locations of the ground sensors illustrate best the concept of the risk tolerance. In Figure 16(a), the locations of the sensors are far away from the locations of the adversaries, having the least exposure, but provide little coverage of the roadway. However, as the risk tolerance factor increases, the sensors move closer to the roadway, providing better coverage despite the increase in the risk of their own detection. Eventually, in Figure 16(d), the sensors almost completely cover the roadway but are relatively close to the observers.

Figure 16. Investigating the risk tolerance factor. The risk tolerance factor, α_0, increases by an increment of 0.5, starting at zero in Figure (a). Note that the locations of the ground sensors (two small black bullets for acoustic sensors and two white triangles for seismic sensors) tend to focus at the roadway with increasing coverage (green color). However, the risk of their own detection by enemies (large black circles on the roadsides) also increases.

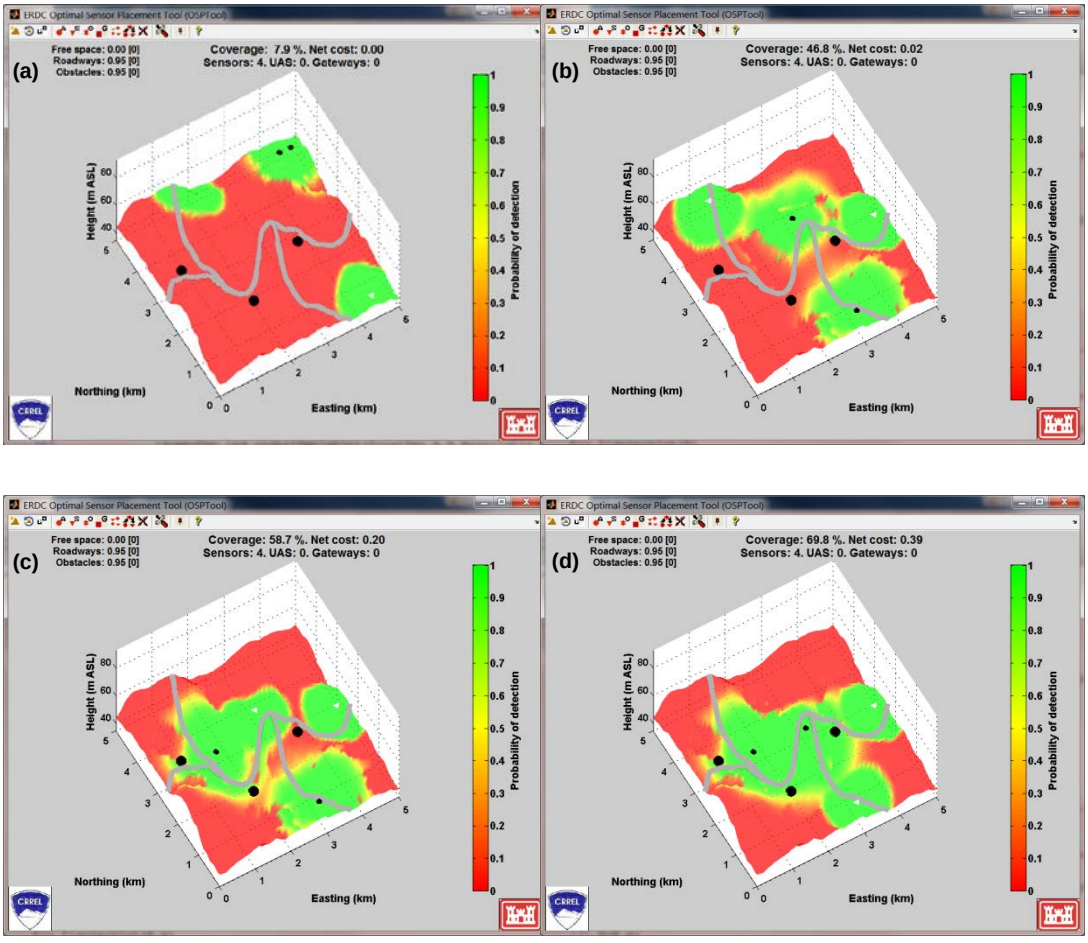
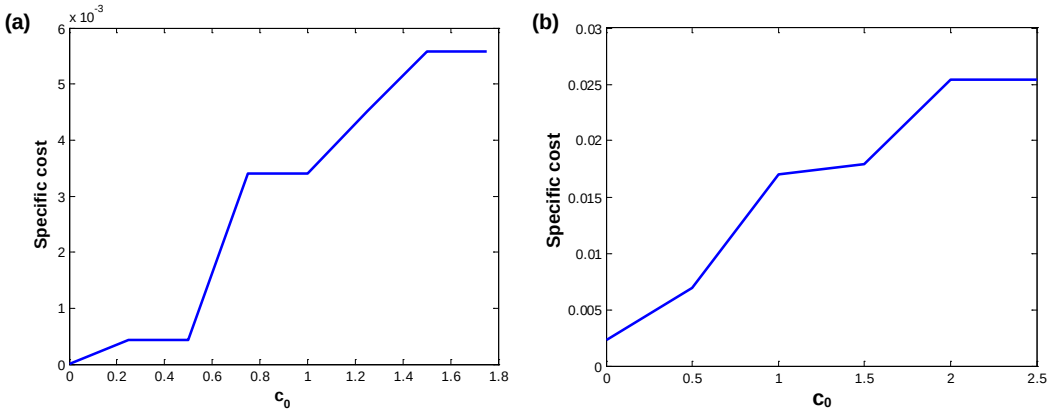


Figure 17 summarizes these results, showing explicitly how the specific cost (per unit coverage) depends on the risk tolerance factor. Figure 17(a) is pertinent to the considered scene whereas Figure 17(b) depicts a similar trend for a completely different scene (the terrain, the object of interest,

and the number of hostile observers). One can see that coverage becomes more and more risky with the increase of the factor. However, small factors result in unacceptably poor coverage. Numerical experiments with different terrain features and objects of interest suggested a trade-off value of $c_0 = 1$, which is set as default in the current program realization.

Figure 17. Dependence of the specific cost (per unit coverage) on the risk tolerance factor, c_0: (a) the scene shown in Figure 9 and (b) a different scene (not shown in the report) with a different terrain, roadway, number and locations of the hostile observers, and the range of UAS altitudes. Note a similar trend in the specific cost.



6 Summary and Conclusion

We can generalize the single-index sensor selection and placement algorithm developed for stationary ground-sensor optimization (formulated in terms of the binary linear programming problem) to support optimization of moving sensors using the set-covering formulation of the routing problem. This approach has several advantages, such as

- it is logically simple and numerically efficient;
- it can incorporate candidate routes specified by different software or a user;
- it can be applied to a point or to extended objects of interest of arbitrary shape;
- it is consistent with flight kinematics (the solutions are smooth);
- it can accommodate a variety of constraints that might be important for a practical use, such as no-fly zones, maximal time of flight, sensor's field of view and required target resolution, oscillatory loitering routing, multiple targets for a single mission, and a preferred view angle of a target;
- it supports optimization of multiple UASs without additional modifications or complications;
- it supports inhomogeneous coverage preferences; and
- it allows joint optimization of stationary and moving sensors in a single mission.

The disadvantages of this approach are that

- there is no optimization of the shape of the route;
- the optimality of the suggested route significantly depends on the number and suitability of the provided admissible routes;
- if an automatic generation of admissible routes is used (as considered in Section 4), the number of admissible routes may be extremely large without good restrictive heuristics.

To maintain compatibility with the stationary sensor framework, the admissible moving-sensor routes should be characterized in terms of coverage and cost. These characteristics depend on time as the sensor moves. In

this report, we characterized the coverage and cost by probability of detection (coverage) and probability of being detected (cost), which enables optimization for covert operations, pertinent to two missions: Mission 1, verify or deny a presence of a target in a certain area, and Mission 2, constantly monitor a target. Efficient calculation of these probabilities in a mathematically exact sense is possible for independent observations. This assumption is likely to be violated for moving sensors. More accurate calculations, suitable for both dependent and independent observations, can be performed in the signal space but require specification of joint multivariate probability distribution functions for signal and noise and numerical evaluation of M-dimensional integrals, where M is the number of route points specifying a route. For typical values of M, needed in practice, the exact calculations of the probabilities may become impractical. Approximate route metrics of maximal and minimal probabilities of detection along the route were suggested for Mission 1 and Mission 2, respectively.

Candidate aircraft routes can be specified either manually or with the use of routing and mission planning software. We developed and used in this study an efficient algorithm for smooth candidate routes. The algorithm provides candidate routes that are not closer than a specified offset distance from the object of interest.

For Mission 1, we considered a specific example scenario of a mountainous roadway requiring reduced audibility at specified locations of hostile observers. Numerical solutions yielded adequate solutions verifying the suitability of the developed routing algorithm and the optimization approach.

However, other approaches are possible. For example, sequential optimization determines first an optimal UAS route and then stationary sensor locations. The route optimization can be specified not by the detection probability but by a completely distinct mission-specific objective function, for example, reflecting an aggregated time at which the target had been observed subject to specified resolution requirements. For this case, a much wider spectrum of route optimization approaches can be applied that are not necessarily compatible with the stationary sensor optimization. Once routing optimization is completed, ground-sensor optimization can be implemented to complete the mission by providing coverage of areas poorly covered by UAS.

References

Ascher, U. M., and L. R. Petzold. 1998. *Computer methods for ordinary differential equations and differential-algebraic equations*. Philadelphia: Society for Industrial and Applied Mathematics.

Abrams, S., P. K. Allen, and K. Tarabanis. 1999. Computing camera viewpoints in an active robot work cell. *The Int. Journal of Robotics Research* 18 (3): 267–285.

Bell, J. E., and P. R. McMullen. 2004. Ant colony optimization techniques for the vehicle routing problem. *Advanced Engineering Informatics* 18 (1): 41–48.

Boscain, U. and B. Piccoli. 2004. *Optimal syntheses for control systems on 2-D manifolds*. New York: Springer Verlag.

Burdic, W. S. 1984. *Underwater acoustic system analysis*. Englewood Cliffs, NJ: Prentice-Hall.

Chin, W., and S. Ntafos. 1991. Watchman routes in simple polygons. *Discrete and Computational Geometry* 6 (1): 9–31.

Dijkstra, E. W. 1959. A note on two problems in connection with graphs. *Numerische Mathematik* 1:269–271.

Dubins, L. 1957. On curves of minimal length with a constraint on average curvature and with prescribed initial and terminal positions and tangents. *American Journal of Mathematics* 79:497–516.

Etemad, K., and R. Chellappa. 1997. Discriminant analysis for recognition of human face images. *Journal of the Optical Society of America* A 14:1724–1734.

Feige, U. 1998. A threshold of ln*n* for approximating set cover. *Journal of the ACM* 45 (4): 634–652.

Gendreau, M., A. Hertz, and G. Laporte. 1994. Tabu search heuristic for the vehicle routing problem. *Management Science* 40 (10): 1276–1290.

Hart, P. E., N. J Nilsson, and B. Raphael. 1968. A formal basis for the heuristic determination of minimum cost paths. *IEEE Transactions on Systems Science and Cybernetics SSC4* 4 (2): 100–107.

Johnson, D. S. 1974. Approximation algorithms for combinatorial problems. *Journal of Computer and System Sciences* 9:256–278.

Johnson, J. 1958. Analysis of image forming systems. In *Proceedings of the Image Intensifier Symposium, AD 220160* (Warfare Electrical Engineering Department, U.S. Army Research and Development Laboratories, Ft. Belvoir, VA): 244–273.

Khardi, S., and L. Abdallah. 2012. Optimization approaches of aircraft flight path reducing noise: comparison of modeling methods. *Applied Acoustics* 73:291–301.

LaValle, S. M. 2006. *Planning algorithms*. Cambridge, UK: Cambridge University Press.

Lee, A. C. 2012. Optimized routing of unmanned aerial systems to address informational gaps in counterinsurgency. MS Thesis, Massachusetts Institute of Technology.

Measure, E. M., D. Knapp, T. Jameson, and A. Butler. 2009. *Automated routing of unmanned aircraft systems (UAS)*. Army Research Laboratory Technical Report ARL-TR-4916. White Sands Missile Range, NM: Army Research Laboratory.

Ombuki B., B. J. Ross, and F. Hanshar. 2006. Multi-objective genetic algorithms for vehicle routing problem with time windows. *Applied Intelligence* 24:17–30.

O'Rourke, J. 1987. *Art gallery theorems and algorithms.* New York: Oxford University Press.

Osman, I. H. 1993. Metastrategy simulated annealing and tabu search algorithms for the vehicle routing problem. *Annals of Operations Research* 41 (4): 421–451.

Phillips, P. J., H. Moon, S. A. Rizvi, and P. J. Rauss. 2000. The FERET evaluation methodology for face-recognition algorithms. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 22 (10): 1090–1104.

Sierksma, G. 2002. *Linear and Integer Programming: Theory and Practice*. 2nd ed. New York: Marcel Dekker.

Toth, P., and D. Vigo, ed. 2001. *The vehicle routing problem*. Philadelphia, PA: Society for Industrial and Applied Mathematics

Vecherin, S. N., D. K. Wilson, and C. L. Pettit. 2010. Optimal placement of multiple types of communicating sensors with availability and coverage redundancy constraints. In *Proceedings of SPIE 7694*, 76940U.

Vecherin, S. N., D. K. Wilson, and C. L. Pettit. 2011. Optimal sensor placement with signal propagation effects and inhomogeneous coverage preferences. *International Journal of Sensor Networks* 9 (2): 107–120.

Wang, P., R. Krishnamurti, and K. Gupta. 2007. Generalized watchman route problem with discrete view cost. In *Proceedings of the 19th Canadian Conference on Computational Geometry (CCCG2007)*, 241–244.

Appendix A: Duration of Flight

In this appendix, the duration of flight, T, is calculated with and without a four-dimensional (space and time) wind field by taking into account a UAS flight control system that counteracts wind and keeps a UAS on its original course.

Let us consider first a case with no wind. We assume the air speed is constant. Let $\mathbf{s}_{i\,i+1} = \mathbf{s}_{i+1} - \mathbf{s}_i$ be a vector connecting two consecutive route points i and $i + 1$ with magnitude $s_{i\,i+1}$ equal to the Euclidian distance between these two points. Then, in the absence of wind, the total time for a route specified by M route points is given by the following equation:

$$T = \frac{1}{V}\sum_{i=1}^{M} s_{i\,i+1}, \tag{A1}$$

where the $(M + 1)$th index indicates the first route point (a closed route), and V is the UAS air speed, which, in this case, coincides with the speed relative to the ground. Distances $s_{i\,i+1}$ may not be all equal to the specified route resolution due to the problematic areas described in Section 4.

In the presence of wind, the calculations are less trivial. The most general wind specification is a four-dimensional, three-component vector wind field, $\mathbf{W}(\mathbf{r},t)$. The resolution in space and time of the wind field is independent of the resolution of a UAS route. We shall assume that the wind field is specified for arbitrary $\mathbf{r}$ and t. Then, as the UAS moves from route point i to the following route point i+1 along a straight line, the wind may change its magnitude and direction continuously. A UAS auto pilot will mitigate these changes by adjusting the direction of the UAS velocity but keeping the airspeed, V, constant, which is assumed to be greater than $W(\mathbf{r},t)$ for any time and location. As a result, a UAS speed along a straight line from i to i+1, denoted $U_{i\,i+1}(\mathbf{r}(s),t)$, becomes variable. Here, s denotes the distance from $\mathbf{s}_i$ towards $\mathbf{s}_{i+1}$ along a straight line, $0 \le s \le s_{i\,i+1}$, and $t_i \le t \le t_{i+1}$. The goal is to determine time $\Delta t_{i\,i+1} = t_{i+1} - t_i$ in these conditions.

Let us represent the wind velocity vector $\mathbf{W}$ and the UAS velocity $\mathbf{V}$ by a sum of two vectors:

$$\mathbf{W}(\mathbf{r},t)=\mathbf{W}_{\parallel}(\mathbf{r},t)+\mathbf{W}_{\perp}(\mathbf{r},t),\quad \mathbf{V}(\mathbf{r},t)=\mathbf{V}_{\parallel}(\mathbf{r},t)+\mathbf{V}_{\perp}(\mathbf{r},t), \tag{A2}$$

where the subscripts $\parallel$ and $\perp$ indicate vectors parallel and perpendicular to $\mathbf{s}_{i\,i+1}$, respectively. Hence:

$$\mathbf{W}_{\parallel}=W_{\parallel}\mathbf{s}^{(0)}_{i\,i+1},\quad W_{\parallel}=\mathbf{W}\cdot\mathbf{s}^{(0)}_{i\,i+1},\quad \mathbf{W}_{\perp}=\mathbf{W}-\mathbf{W}_{\parallel}, \tag{A3}$$

where $\mathbf{s}^{(0)}_{i\,i+1}=\mathbf{s}_{i\,i+1}/s_{i\,i+1}$ is a unit vector in the direction of $\mathbf{s}_{i\,i+1}$. To counteract the transverse component and to keep the UAS flying in the direction $\mathbf{s}_{i\,i+1}$, a UAS control unit will set $\mathbf{V}_{\perp}=-\mathbf{W}_{\perp}$, and thus

$$V_{\parallel}=\left(V^2-V_{\perp}^{\,2}\right)^{1/2}=\left(V^2-W^2+W_{\parallel}^{\,2}\right)^{1/2}. \tag{A4}$$

Therefore, the resulting UAS speed along $\mathbf{s}_{i\,i+1}$ is

$$\begin{aligned} &U(\mathbf{r},t)=W_{\parallel}+V_{\parallel}=\mathbf{W}\cdot\mathbf{s}^{(0)}_{i\,i+1}+\left(V^2-W^2+W_{\parallel}^{\,2}\right)^{1/2},\\ &\mathbf{r}(s)=\mathbf{s}_i+s\,\mathbf{s}^{(0)}_{i\,i+1}. \end{aligned} \tag{A5}$$

The problem of the calculation of the time interval $\Delta t_{i\,i+1}$ can now be formulated in conventional terms of the ordinary differential equation taking the travel length s as an independent variable (for convenience, to guarantee the arrival to the next route point exactly) and considering unknown time t as function of the travelled distance, $t = t(s)$:

$$\frac{dt}{ds}=\frac{1}{U(s,t)},\quad t(0)=t_i. \tag{A6}$$

Equation (A6) can be solved numerically for $0\le s\le s_{i\,i+1}$ by standard numerical techniques, for example, by the fourth-order Runge-Kutta method or one of its variations (Ascher and Petzold 1998).

The total travel time along the route is found by the summation of all $\Delta t_{i\,i+1}$.

The following algorithm with a progressively decreasing integration step size can be used to calculate $\Delta t_{i\,i+1}$.

Algorithm 3. Travel time $\Delta t_{i\,i+1}$ with variable speed.

Inputs: ($\mathbf{s}_i$, $\mathbf{s}_{i+1}$) – three-dimensional space vectors indicating locations of the ith and (i+1)th route points; $U(s,t)$ – UAS speed from Equation (A.5); t_i – time at which UAS was at the $\mathbf{s}_i$ location, ε – relative threshold of the integration determining the accuracy of the sought time interval Δt.

Output: Δt.

1 calculate distance $s_{i\,i+1} = |\,\mathbf{s}_{i+1} - \mathbf{s}_i|$;
2 set the initial integration step $ds = 0.5\, s_{i\,i+1}$;
3 calculate initial travel-time interval Δt_0 by solving Equation (A.6);
4 initialize the relative error: $e = \varepsilon + 1$;
5 WHILE $e > \varepsilon$;
6 reduce the integration step: $ds = ds/2$;
7 calculate a new travel-time interval Δt;
8 calculate the relative error: $e = |\,\Delta t - \Delta t_0|/\,\Delta t_0$
9 $\Delta t_0 = \Delta t$;
10 END WHILE

REPORT DOCUMENTATION PAGE

Form Approved
OMB No. 0704-0188

1. REPORT DATE (DD-MM-YYYY)	2. REPORT TYPE	3. DATES COVERED (From - To)
17-09-2014	Technical Report/Final	

4. TITLE AND SUBTITLE
Coordinated Optimization of Aircraft Routes and Locations of Ground Sensors

5a. CONTRACT NUMBER

5b. GRANT NUMBER

5c. PROGRAM ELEMENT

6. AUTHOR(S)
Sergey N. Vecherin , D. Keith Wilson , and Chris L. Pettit

5d. PROJECT NUMBER

5e. TASK NUMBER

5f. WORK UNIT NUMBER

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)
U.S. Army Engineer Research and Development Center (ERDC)
Cold Regions Research and Engineering Laboratory (CRREL)
72 Lyme Road,
Hanover, NH 03755-1290

8. PERFORMING ORGANIZATION REPORT NUMBER
ERDC/CRREL TR-14-20

9. SPONSORING / MONITORING AGENCY NAME(S) AND ADDRESS(ES)
Headquarters, U.S. Army Corps of Engineers
Washington, DC 20314-1000

10. SPONSOR/MONITOR'S ACRONYM(S)

11. SPONSOR/MONITOR'S REPORT NUMBER(S)

12. DISTRIBUTION / AVAILABILITY STATEMENT
Approved for public release; distribution is unlimited.

13. SUPPLEMENTARY NOTES

14. ABSTRACT

In this report, an optimal sensor placement tool, developed for determining near-optimal configurations of stationary ground sensors, is generalized to support aircraft routing. This generalization requires characterizing candidate aircraft routes in terms of cost and coverage. Cost can reflect a variety of disincentives, not necessarily monetary—for example, a probability of aircraft to be heard on the ground. Several metrics for moving sensor platforms were considered to adequately characterize cost and coverage. The generalized algorithm can be applied to such practical problems as determining the optimal combination of routes for multiple aircraft operations, optimizing routes to supplement ground-sensor coverage, optimizing ground sensors to cover blind spots of aircraft coverage, and simultaneously optimizing static and moving sensor platforms. An example problem that this report considers in detail is unmanned aircraft system (UAS) routing for verification of roadway security while minimizing UAS audibility at specified locations on the ground.

15. SUBJECT TERMS
Binary linear programming, Joint optimization of UAS and ground sensors, Multiple UAS optimization, Optimal routes, Optimal sensor placement, Optimal sensor selection, Set covering optimization, UAS route optimization

16. SECURITY CLASSIFICATION OF:			17. LIMITATION OF ABSTRACT	18. NUMBER OF PAGES	19a. NAME OF RESPONSIBLE PERSON
a. REPORT	b. ABSTRACT	c. THIS PAGE			19b. TELEPHONE NUMBER (include area code)
Unclassified	Unclassified	Unclassified	SAR	55	